

## **NGÂN HÀNG CÂU HỎI THI THỰC HÀNH - VẤN ĐÁP**

**Tên học phần: NHẬP MÔN CÔNG NGHỆ PHẦN MỀM    Mã học phần:.....**

**Ngành đào tạo: CÔNG NGHỆ THÔNG TIN    Trình độ đào tạo: Đại học chính quy**

### **Cấu trúc đề**

Mỗi đề có hai phần:

- Phần lí thuyết 2,5 điểm, do giáo viên hỏi ngay khi chấm phần bài tập của sinh viên
- Phần bài tập thực hành: 7,5 điểm, có dạng một bài tập hoàn chỉnh. Mỗi đề có 5 câu hỏi liên quan đến các bước của công nghệ phần mềm. Mỗi câu 1,5 điểm.

Thời gian chung cho tất cả các đề là 60 phút.

*Ngân hàng câu hỏi thi này đã được thông qua bộ môn và nhóm cán bộ giảng dạy học phần.*

		<i>Hà Nội, ngày    tháng    năm</i>
<b>Trưởng khoa</b>	<b>Trưởng bộ môn</b>	<b>Giảng viên chủ trì biên soạn</b>

PGS.TS. Từ Minh Phương

TS. Nguyễn Duy Phương

TS. Nguyễn Mạnh Hùng

## **Phần lí thuyết**

- Phần lí thuyết được hỏi ngay khi giáo viên chấm phần bài tập cho sinh viên
- Điểm của phần lí thuyết là 2,5 điểm
- Giáo viên có thể hỏi một số câu hỏi khác nhau, tùy thuộc khả năng trả lời của sinh viên

## **Câu hỏi lý thuyết Kỹ nghệ phần mềm**

1. Thế nào là corrective maintenance?
2. Thế nào là adaptive maintenance?
3. Thế nào là perfective maintenance?
4. Thế nào là refactoring?
5. Thế nào là "from scratch"?
6. Thế nào là moving target problem?
7. Thế nào là regression fault?
8. Thế nào là một episode?
9. Thế nào là một iteration?
10. Thế nào là một increasement?
11. Thế nào là một artifact?
12. Thế nào là portability?
13. Thế nào là reuseability?
14. Thế nào là milestone?
15. Thế nào là một story?
16. Thế nào là refactoring?
17. Thế nào là concept exploration?
18. Thế nào là business model?
19. Thế nào là traceability?
20. Thế nào là egoless programming?
21. Thế nào là PM?
22. Thế nào là technical leader?
23. Thế nào là programming secretary?
24. Thế nào là backup programmer?
25. Thế nào là supper programmer?
26. Thế nào là một bản thiết kế còn ommision?
27. Thế nào là một bản thiết kế còn contradiction?
28. Thế nào là một phần mềm COTS?
29. Thế nào là SPMP?
30. Thế nào là alpha release?
31. Thế nào là beta release?
32. Thế nào là process?
33. Thế nào là workflow?
34. Luật Miller trong CNPM nói gì?
35. Luật Brooks trong CNPM nói gì?
36. Luật Dijkstra trong CNPM nói gì?
37. Verification và Validation (V&V) là gì?
38. Thế nào là inspection?
39. Thế nào walkthrough?
40. Thế nào là một moderator trong nhóm inspection?
41. Thế nào là một recorder trong nhóm inspection?
42. Mô hình CMM là gì?
43. Thế nào là test performance?
44. Thế nào là test robustness?
45. Thế nào là coin of uncertainty?
46. Thế nào là norminal effort?
47. Thế nào là phần mềm organic?

48. Thế nào là phần mềm embeded?
49. Thế nào là phần mềm semi-detached?
50. Thế nào là TCF?
51. Thế nào là UFP?
52. Thế nào là flow trong FFP?
53. Thế nào là process trong FFP?
54. Tại sao không có pha kiểm thử?
55. Tại sao không có pha làm tài liệu?
56. Tại sao không có pha lập kế hoạch?
57. Nếu không áp dụng các mô hình vòng đời phần mềm thì có phát triển được phần mềm không? Tại sao?
58. Tại sao người ta phải dùng nhiều mô hình vòng đời khác nhau để phát triển phần mềm?
59. Nêu ưu điểm, nhược điểm của mô hình vòng đời phần mềm kiểu thác nước?
60. Mô hình vòng đời phần mềm kiểu thác nước thì phù hợp với những dự án có đặc điểm gì?
61. Nêu ưu điểm, nhược điểm của mô hình vòng đời phần mềm kiểu bản mẫu nhanh?
62. Mô hình vòng đời phần mềm kiểu bản mẫu nhanh thì phù hợp với những dự án có đặc điểm gì?
63. Nêu ưu điểm, nhược điểm của mô hình vòng đời phần mềm kiểu lặp và tăng trưởng?
64. Mô hình vòng đời phần mềm kiểu lặp và tăng trưởng thì phù hợp với những dự án có đặc điểm gì?
65. Nêu ưu điểm, nhược điểm của mô hình vòng đời phần mềm kiểu xoắn ốc?
66. Mô hình vòng đời phần mềm kiểu xoắn ốc thì phù hợp với những dự án có đặc điểm gì?
67. Nêu ưu điểm, nhược điểm của mô hình vòng đời phần mềm kiểu tiến trình linh hoạt?
68. Mô hình vòng đời phần mềm kiểu tiến trình linh hoạt thì phù hợp với những dự án có đặc điểm gì?
69. Trong mô hình tiến trình liên hoạt, luôn có đại diện của khách hàng trong nhóm phát triển thì có ưu điểm gì?
70. Nêu ưu điểm, nhược điểm của mô hình nhóm code bình đẳng?
71. Mô hình nhóm code bình đẳng thì phù hợp với những dự án có đặc điểm gì?
72. Nêu ưu điểm, nhược điểm của mô hình nhóm code có chef?
73. Mô hình nhóm code có chef thì phù hợp với những dự án có đặc điểm gì?
74. Nêu ưu điểm, nhược điểm của kỹ thuật pair programming?
75. Kỹ thuật pair programming thì phù hợp với những dự án có đặc điểm gì?
76. Nêu ưu điểm, nhược điểm của kỹ thuật time boxing?
77. Nêu ưu điểm, nhược điểm của kỹ thuật stand up meeting?
78. Nêu ưu điểm, nhược điểm của phương pháp ước lượng phần mềm bằng LOC?
79. Nêu ưu điểm, nhược điểm của phương pháp ước lượng phần mềm bằng FFP?
80. Nêu ưu điểm, nhược điểm của phương pháp ước lượng phần mềm bằng Function Point?
81. Nêu ưu điểm, nhược điểm của phương pháp ước lượng phần mềm bằng COCOMO?
82. Tại sao trong mô hình tiến trình linh hoạt, không cần có pha đặc tả?
83. Tại sao trong nhóm walkthrough và inspection, luôn phải có đại diện của workflow tiếp theo?
84. Nếu nhóm SQA phát hiện ra ít lỗi, thì có thể kết luận nhóm code giỏi hay nhóm SQA kém? Tại sao?
85. Tại sao nói inspection và walkthrough là hướng tài liệu, mà không phải hướng vào người tham gia?
86. Quality assurance thì khác gì với testing?
87. Tại sao nói function point chịu ảnh hưởng chủ quan của các chuyên gia?
88. COCOMO tính đến nhiều tiêu chí hơn hay là function point? Giải thích?
89. SW development multiplier của COCOMO thì khác gì TCF của function point?

90. TCF của function point thì khác gì hằng số b của FFP?
91. Tại sao nguyên lý Dijkstra lại đúng?
92. Tại sao luật Brook lại đúng?
93. Người ta áp dụng luật Miller trong CNPM như thế nào?
94. Phát triển phần mềm thì khác gì sản xuất phần mềm?
95. Test trường hợp sai kiểu dữ liệu đầu vào thì thuộc thể loại test gì?

## **Phần bài tập**

- Mỗi đề thi trong phần bài tập cần làm trong thời gian dự kiến là 60 phút
- Điểm phần bài tập thực hành là 7,5 điểm. Mỗi câu hỏi con có điểm bằng nhau: 1,5 điểm/câu

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 01**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý thư viện, được mô tả như sau:

- Mỗi đầu sách (Mã, tên, tác giả, năm xuất bản, giá bìa, số lượng, mã vạch, mô tả) có thể được mượn nhiều lần khác nhau bởi nhiều bạn đọc khác nhau
- Mỗi bạn đọc có một thẻ bạn đọc chứa mã, tên, ngày sinh, địa chỉ, số điện thoại, mã vạch của bạn đọc đó
- Mỗi lần mượn được mượn tối đa 5 quyển sách, và tổng số sách đang mượn bởi một người cũng không được quá 5 quyển
- Thời gian tối đa mượn 1 quyển sách là 1 tháng kể từ ngày mượn quyển đó, nếu trả sau thời hạn này thì sẽ bị phạt 20% giá trị bìa sách.
- Mỗi lần trả sách có thể trả một phần hoặc toàn bộ số lượng sách đang mượn
- Khi mượn sách mới, thủ thư vẫn xem được danh sách các sách mà một độc giả đã mượn và trả rồi hoặc chưa trả trước đây.

Anh/chị hãy thực hiện modul "***Quản lý sách***" với các bước sau đây: cho phép quản lý (QL) thực hiện thêm, sửa, xóa thông tin sách với mô tả chi tiết nghiệp vụ: QL chọn menu quản lý sách → trang quản lý hiện ra → QL chọn chức năng sửa thông tin sách → giao diện tìm sách theo tên hiện ra → QL nhập tên sách và click tìm kiếm → danh sách các sách có tên chứa từ khóa hiện ra → QL chọn sửa một sách → giao diện sửa sách hiện ra với các thông tin của sách đã chọn → QL nhập một số thông tin thay đổi và click cập nhật → hệ thống lưu thông tin vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 02**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý thư viện, được mô tả như sau:

- Mỗi đầu sách (Mã, tên, tác giả, năm xuất bản, giá bìa, số lượng, mã vạch, mô tả) có thể được mượn nhiều lần khác nhau bởi nhiều bạn đọc khác nhau
- Mỗi bạn đọc có một thẻ bạn đọc chứa mã, tên, ngày sinh, địa chỉ, số điện thoại, mã vạch của bạn đọc đó
- Mỗi lần mượn được mượn tối đa 5 quyển sách, và tổng số sách đang mượn bởi một người cũng không được quá 5 quyển
- Thời gian tối đa mượn 1 quyển sách là 1 tháng kể từ ngày mượn quyển đó, nếu trả sau thời hạn này thì sẽ bị phạt 20% giá trị bìa sách.
- Mỗi lần trả sách có thể trả một phần hoặc toàn bộ số lượng sách đang mượn
- Khi mượn sách mới, thủ thư vẫn xem được danh sách các sách mà một độc giả đã mượn và trả rồi hoặc chưa trả trước đây.

Anh/chị hãy thực hiện modul "***Quản lý việc mượn sách***" với các bước sau đây: Nhân viên chọn menu cho mượn sách → quét thẻ độc giả để lấy thông tin độc giả → thông tin chi tiết độc giả hiện lên + danh sách các sách mượn chưa trả + danh sách sách mượn đã trả → nhân viên quét lần lượt các sách được chọn mượn → danh sách sách mượn được bổ sung thêm cho đến khi hết sách chọn mượn (hoặc tối đa 5 quyển) thì submit → in ra phiếu mượn chứa mã, tên, mã vạch độc giả, mã vạch phiếu mượn, và danh sách sách còn mượn, mỗi đầu sách trên một dòng: mã, tên sách, tác giả, mã vạch, ngày mượn, ngày phải trả và dòng cuối cùng ghi tổng số sách đang mượn.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này



**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 03**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý thư viện, được mô tả như sau:

- Mỗi đầu sách (Mã, tên, tác giả, năm xuất bản, giá bìa, số lượng, mã vạch, mô tả) có thể được mượn nhiều lần khác nhau bởi nhiều bạn đọc khác nhau
- Mỗi bạn đọc có một thẻ bạn đọc chứa mã, tên, ngày sinh, địa chỉ, số điện thoại, mã vạch của bạn đọc đó
- Mỗi lần mượn được mượn tối đa 5 quyển sách, và tổng số sách đang mượn bởi một người cũng không được quá 5 quyển
- Thời gian tối đa mượn 1 quyển sách là 1 tháng kể từ ngày mượn quyển đó, nếu trả sau thời hạn này thì sẽ bị phạt 20% giá trị bìa sách.
- Mỗi lần trả sách có thể trả một phần hoặc toàn bộ số lượng sách đang mượn
- Khi mượn sách mới, thủ thư vẫn xem được danh sách các sách mà một độc giả đã mượn và trả rồi hoặc chưa trả trước đây.

Anh/chị hãy thực hiện modul "***Quản lý việc trả sách***" với các bước sau đây: Nhân viên chọn menu trả sách → quét thẻ độc giả để lấy thông tin độc giả → thông tin chi tiết độc giả hiện lên + danh sách các sách mượn chưa trả + danh sách sách mượn đã trả → nhân viên quét lần lượt các sách được trả → danh sách sách đang mượn được rút ngắn cho đến khi hết sách mượn (hoặc hết số sách độc giả đem đến trả) thì submit → in ra phiếu mượn (nếu còn sách mượn) chứa mã, tên, mã vạch độc giả, mã vạch phiếu mượn, và danh sách sách còn mượn, mỗi đầu sách trên một dòng: mã, tên sách, tác giả, mã vạch, ngày mượn, ngày phải trả và dòng cuối cùng ghi tổng số sách đang mượn + phiếu phạt (nếu bị phạt) chứa mã, tên, mã vạch độc giả, mã vạch phiếu mượn, và danh sách sách trả mượn bị phạt, mỗi đầu sách trên một dòng: mã, tên sách, tác giả, mã vạch, ngày mượn, ngày phải trả, ngày trả, số tiền phạt và dòng cuối cùng ghi tổng số tiền phạt

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 04**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý thư viện, được mô tả như sau:

- Mỗi đầu sách (Mã, tên, tác giả, năm xuất bản, giá bìa, số lượng, mã vạch, mô tả) có thể được mượn nhiều lần khác nhau bởi nhiều bạn đọc khác nhau
- Mỗi bạn đọc có một thẻ bạn đọc chứa mã, tên, ngày sinh, địa chỉ, số điện thoại, mã vạch của bạn đọc đó
- Mỗi lần mượn được mượn tối đa 5 quyển sách, và tổng số sách đang mượn bởi một người cũng không được quá 5 quyển
- Thời gian tối đa mượn 1 quyển sách là 1 tháng kể từ ngày mượn quyển đó, nếu trả sau thời hạn này thì sẽ bị phạt 20% giá trị bìa sách.
- Mỗi lần trả sách có thể trả một phần hoặc toàn bộ số lượng sách đang mượn
- Khi mượn sách mới, thủ thư vẫn xem được danh sách các sách mà một độc giả đã mượn và trả rồi hoặc chưa trả trước đây.

Anh/chị hãy thực hiện modul "***Thống kê sách mượn nhiều***" với các bước sau đây: Nhân viên chọn menu thống kê → chọn thống kê sách mượn nhiều → nhập khoảng thời gian (bắt đầu - kết thúc) → danh sách sách mượn nhiều nhất được hiển thị theo thứ tự số lượt mượn từ nhiều đến ít, mỗi dòng chứa: mã, tên sách, tác giả, mã vạch, tổng số lượt mượn. NV click vào 1 dòng của 1 sách thì hiện lên danh sách chi tiết những lần độc giả nào mượn quyển sách đấy.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 05**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý thư viện, được mô tả như sau:

- Mỗi đầu sách (Mã, tên, tác giả, năm xuất bản, giá bìa, số lượng, mã vạch, mô tả) có thể được mượn nhiều lần khác nhau bởi nhiều bạn đọc khác nhau
- Mỗi bạn đọc có một thẻ bạn đọc chứa mã, tên, ngày sinh, địa chỉ, số điện thoại, mã vạch của bạn đọc đó
- Mỗi lần mượn được mượn tối đa 5 quyển sách, và tổng số sách đang mượn bởi một người cũng không được quá 5 quyển
- Thời gian tối đa mượn 1 quyển sách là 1 tháng kể từ ngày mượn quyển đó, nếu trả sau thời hạn này thì sẽ bị phạt 20% giá trị bìa sách.
- Mỗi lần trả sách có thể trả một phần hoặc toàn bộ số lượng sách đang mượn
- Khi mượn sách mới, thủ thư vẫn xem được danh sách các sách mà một độc giả đã mượn và trả rồi hoặc chưa trả trước đây.

Anh/chị hãy thực hiện modul "**Thống kê độc giả mượn nhiều**" với các bước sau đây: Nhân viên chọn menu thống kê → chọn thống kê độc giả mượn nhiều → nhập khoảng thời gian (bắt đầu - kết thúc) → danh sách độc giả đã mượn nhiều nhất được hiển thị theo thứ tự số lượng sách mượn từ nhiều đến ít, mỗi dòng chứa: mã, tên, ngày sinh, địa chỉ độc giả, tổng số lượng sách đã mượn. NV click vào 1 dòng của 1 độc giả thì hiện lên chi tiết các phiếu mượn với thông tin ngày mượn, tổng số sách của từng lần mượn.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 06**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý kết quả học tập của sinh viên theo tín chỉ, được mô tả như sau:

- Mỗi sinh viên (Mã SV, mật khẩu, tên, ngày sinh, khóa, quê quán, địa chỉ) được phép đăng kí tối thiểu 10 tín chỉ/học kì và tối đa 15 tín chỉ/học kì
- Mỗi sinh viên được đăng kí nhiều môn học (mã môn, tên môn, số tín chỉ)
- Mỗi môn học có thể có nhiều môn học yêu cầu sinh viên phải hoàn thành trước đó thì mới được đăng kí
- Mỗi môn học có thể có nhiều lớp học phần (mã lớp, tên lớp, số sv tối đa, phòng học, khung giờ học cố định trong tuần)
- Sinh viên không được phép đăng kí học hai lớp có trùng buổi học
- Với mỗi môn học, một sinh viên chỉ được đăng kí vào 1 lớp xác định
- Kết quả của sinh viên (điểm thành phần số 1, số 2, số 3, điểm thi, điểm cuối cùng= $x\%$  số1+ $y\%$  số2 +  $z\%$  số3 +  $w\%$  điểm thi) được lưu theo từng môn học
- Điểm trung bình của sinh viên trong học kì được tính bằng trung bình có trọng số là số tín chỉ từng môn học

Anh/chị hãy thực hiện modul "**Quản lý thông tin sinh viên**" với các bước sau đây: cho phép quản lí (QL) thực hiện thêm, sửa, xóa thông tin sinh viên với mô tả chi tiết nghiệp vụ: QL chọn menu quản lí sinh viên → trang quản lí hiện ra → QL chọn chức năng sửa thông tin sinh viên → giao diện tìm sinh viên theo tên hiện ra → QL nhập tên sinh viên và click tìm kiếm → danh sách các sinh viên có tên chứa từ khóa hiện ra → QL chọn sửa một sinh viên → giao diện sửa sinh viên hiện ra với các thông tin của sinh viên đã chọn → QL nhập một số thông tin thay đổi và click cập nhật → hệ thống lưu thông tin vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 07**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý kết quả học tập của sinh viên theo tín chỉ, được mô tả như sau:

- Mỗi sinh viên (Mã SV, mật khẩu, tên, ngày sinh, khóa, quê quán, địa chỉ) được phép đăng kí tối thiểu 10 tín chỉ/học kì và tối đa 15 tín chỉ/học kì
- Mỗi sinh viên được đăng kí nhiều môn học (mã môn, tên môn, số tín chỉ)
- Mỗi môn học có thể có nhiều môn học yêu cầu sinh viên phải hoàn thành trước đó thì mới được đăng kí
- Mỗi môn học có thể có nhiều lớp học phần (mã lớp, tên lớp, số sv tối đa, phòng học, khung giờ học cố định trong tuần)
- Sinh viên không được phép đăng kí học hai lớp có trùng buổi học
- Với mỗi môn học, một sinh viên chỉ được đăng kí vào 1 lớp xác định
- Kết quả của sinh viên (điểm thành phần số 1, số 2, số 3, điểm thi, điểm cuối cùng= $x\%$  số1+ $y\%$  số2 +  $z\%$  số3 +  $w\%$  điểm thi) được lưu theo từng môn học
- Điểm trung bình của sinh viên trong học kì được tính bằng trung bình có trọng số là số tín chỉ từng môn học

Anh/chị hãy thực hiện modul "**Lên lịch học cho lớp học phần**" với các bước sau đây: QL chọn menu lên lịch học cho lớp học phần → giao diện lên lịch học hiện ra với các ô số chọn môn học, lớp học phần, phòng học, khung giờ, nút xác nhận → QL click chọn môn học từ danh sách sổ xuống → Danh sách lớp học phần của môn học học được cập nhật → QL click chọn thêm 1 lớp học phần của môn học → click chọn phòng học từ danh sách phòng học sổ xuống + click chọn khung giờ trong tuần từ danh sách khung giờ sổ xuống + click xác nhận → Hệ thống lưu lịch học vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 08**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý kết quả học tập của sinh viên theo tín chỉ, được mô tả như sau:

- Mỗi sinh viên (Mã SV, mật khẩu, tên, ngày sinh, khóa, quê quán, địa chỉ) được phép đăng kí tối thiểu 10 tín chỉ/học kì và tối đa 15 tín chỉ/học kì
- Mỗi sinh viên được đăng kí nhiều môn học (mã môn, tên môn, số tín chỉ)
- Mỗi môn học có thể có nhiều môn học yêu cầu sinh viên phải hoàn thành trước đó thì mới được đăng kí
- Mỗi môn học có thể có nhiều lớp học phần (mã lớp, tên lớp, số sv tối đa, phòng học, khung giờ học cố định trong tuần)
- Sinh viên không được phép đăng kí học hai lớp có trùng buổi học
- Với mỗi môn học, một sinh viên chỉ được đăng kí vào 1 lớp xác định
- Kết quả của sinh viên (điểm thành phần số 1, số 2, số 3, điểm thi, điểm cuối cùng= $x\%$  số1+ $y\%$  số2 +  $z\%$  số3 +  $w\%$  điểm thi) được lưu theo từng môn học
- Điểm trung bình của sinh viên trong học kì được tính bằng trung bình có trọng số là số tín chỉ từng môn học

Anh/chị hãy thực hiện modul "**Nhập điểm theo lớp học phần**" với mô tả chi tiết nghiệp vụ: giáo viên hoặc giáo vụ (GV) chọn chức năng nhập điểm → giao diện hiện ra danh sách các môn học do GV dạy → GV click chọn 1 môn học → giao diện hiện ra danh sách các lớp học phần của môn học đã chọn do GV dạy → GV click chọn 1 lớp học phần → Giao diện hiện lên danh sách các sinh viên trong lớp học phần, mỗi SV trên 1 dòng với các cột điểm thành phần và cột điểm thi → GV nhập đầy đủ các đầu điểm của các SV + click xác nhận → Hệ thống lưu vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 09**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý kết quả học tập của sinh viên theo tín chỉ, được mô tả như sau:

- Mỗi sinh viên (Mã SV, mật khẩu, tên, ngày sinh, khóa, quê quán, địa chỉ) được phép đăng kí tối thiểu 10 tín chỉ/học kì và tối đa 15 tín chỉ/học kì
- Mỗi sinh viên được đăng kí nhiều môn học (mã môn, tên môn, số tín chỉ)
- Mỗi môn học có thể có nhiều môn học yêu cầu sinh viên phải hoàn thành trước đó thì mới được đăng kí
- Mỗi môn học có thể có nhiều lớp học phần (mã lớp, tên lớp, số sv tối đa, phòng học, khung giờ học cố định trong tuần)
- Sinh viên không được phép đăng kí học hai lớp có trùng buổi học
- Với mỗi môn học, một sinh viên chỉ được đăng kí vào 1 lớp xác định
- Kết quả của sinh viên (điểm thành phần số 1, số 2, số 3, điểm thi, điểm cuối cùng= $x\%$  số1+ $y\%$  số2 +  $z\%$  số3 +  $w\%$  điểm thi) được lưu theo từng môn học
- Điểm trung bình của sinh viên trong học kì được tính bằng trung bình có trọng số là số tín chỉ từng môn học

Anh/chị hãy thực hiện modul "**Đăng kí học**" với các bước sau đây: Học sinh đăng nhập → chọn menu đăng kí tín chỉ cho học kì mới → trang đăng kí hiện ra → sinh viên chọn môn học trong danh sách môn học + chọn lớp trong danh sách các lớp (và giảng viên đi kèm) tương ứng với môn học → nếu thỏa mãn các ràng buộc nêu trên thì thông báo thành công + in ra phiếu đăng kí cho sinh viên: mã SV, tên SV, khóa học, học kì+danh sách các môn học đã đăng kí, mỗi môn có: mã MH, tên MH, số tín chỉ, giờ học, giảng viên

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 10**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý kết quả học tập của sinh viên theo tín chỉ, được mô tả như sau:

- Mỗi sinh viên (Mã SV, mật khẩu, tên, ngày sinh, khóa, quê quán, địa chỉ) được phép đăng kí tối thiểu 10 tín chỉ/học kì và tối đa 15 tín chỉ/học kì
- Mỗi sinh viên được đăng kí nhiều môn học (mã môn, tên môn, số tín chỉ)
- Mỗi môn học có thể có nhiều môn học yêu cầu sinh viên phải hoàn thành trước đó thì mới được đăng kí
- Mỗi môn học có thể có nhiều lớp học phần (mã lớp, tên lớp, số sv tối đa, phòng học, khung giờ học cố định trong tuần)
- Sinh viên không được phép đăng kí học hai lớp có trùng buổi học
- Với mỗi môn học, một sinh viên chỉ được đăng kí vào 1 lớp xác định
- Kết quả của sinh viên (điểm thành phần số 1, số 2, số 3, điểm thi, điểm cuối cùng= $x\%$  số1+ $y\%$  số2 +  $z\%$  số3 +  $w\%$  điểm thi) được lưu theo từng môn học
- Điểm trung bình của sinh viên trong học kì được tính bằng trung bình có trọng số là số tín chỉ từng môn học

Anh/chị hãy thực hiện modul "**Xem TKB của sinh viên**" với các bước sau đây: SV chọn menu xem TKB → Giao diện xem TKB hiện lên với phía trên là ô chọn các cách xem TKB theo: tuần, học kỳ → SV chọn xem theo tuần → Phía dưới cập nhật hiển thị thời khóa biểu theo tuần hiện tại của SV: 1 bảng có 7 cột tương ứng 7 ngày, 6 hàng tương ứng 6 kíp học cho mỗi ngày. Trong mỗi ô của bảng hiển thị tên môn học, nhóm môn học, và tên phòng học tương ứng với khung giờ đó.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này



**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 11**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý kết quả học tập của sinh viên theo tín chỉ, được mô tả như sau:

- Mỗi sinh viên (Mã SV, mật khẩu, tên, ngày sinh, khóa, quê quán, địa chỉ) được phép đăng kí tối thiểu 10 tín chỉ/học kì và tối đa 15 tín chỉ/học kì
- Mỗi sinh viên được đăng kí nhiều môn học (mã môn, tên môn, số tín chỉ)
- Mỗi môn học có thể có nhiều môn học yêu cầu sinh viên phải hoàn thành trước đó thì mới được đăng kí
- Mỗi môn học có thể có nhiều lớp học phần (mã lớp, tên lớp, số sv tối đa, phòng học, khung giờ học cố định trong tuần)
- Sinh viên không được phép đăng kí học hai lớp có trùng buổi học
- Với mỗi môn học, một sinh viên chỉ được đăng kí vào 1 lớp xác định
- Kết quả của sinh viên (điểm thành phần số 1, số 2, số 3, điểm thi, điểm cuối cùng= $x\%$  số1+ $y\%$  số2 +  $z\%$  số3 +  $w\%$  điểm thi) được lưu theo từng môn học
- Điểm trung bình của sinh viên trong học kì được tính bằng trung bình có trọng số là số tín chỉ từng môn học

Anh/chị hãy thực hiện modul "**Thống kê sinh viên khá giỏi**" với các bước sau đây: quản lí đăng nhập → chọn menu thống kê → chọn thống kê sinh viên giỏi → trang kết quả hiện ra danh sách SV: mã SV, tên SV, khóa học, học kì, tổng số tín chỉ đã học trong học kì, điểm trung bình môn cuối học kì, sắp xếp theo điểm trung bình cả học kì, từ cao đến thấp. NV click vào 1 dòng của 1 SV thì hiện lên chi tiết bảng điểm từng môn học mà SV đã học trong học kì.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 12**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý kết quả học tập của sinh viên theo tín chỉ, được mô tả như sau:

- Mỗi sinh viên (Mã SV, mật khẩu, tên, ngày sinh, khóa, quê quán, địa chỉ) được phép đăng kí tối thiểu 10 tín chỉ/học kì và tối đa 15 tín chỉ/học kì
- Mỗi sinh viên được đăng kí nhiều môn học (mã môn, tên môn, số tín chỉ)
- Mỗi môn học có thể có nhiều môn học yêu cầu sinh viên phải hoàn thành trước đó thì mới được đăng kí
- Mỗi môn học có thể có nhiều lớp học phần (mã lớp, tên lớp, số sv tối đa, phòng học, khung giờ học cố định trong tuần)
- Sinh viên không được phép đăng kí học hai lớp có trùng buổi học
- Với mỗi môn học, một sinh viên chỉ được đăng kí vào 1 lớp xác định
- Kết quả của sinh viên (điểm thành phần số 1, số 2, số 3, điểm thi, điểm cuối cùng= $x\% \text{ số1} + y\% \text{ số2} + z\% \text{ số3} + w\% \text{ điểm thi}$ ) được lưu theo từng môn học
- Điểm trung bình của sinh viên trong học kì được tính bằng trung bình có trọng số là số tín chỉ từng môn học

Anh/chị hãy thực hiện modul "***Thống kê môn học theo tỉ lệ sinh viên qua***" với các bước sau đây:  
quản lý đăng nhập → chọn menu thống kê → chọn thống kê môn học của từng giáo viên dạy theo tỉ lệ SV qua môn → trang kết quả hiện ra danh sách môn học: mã MH, tên MH, số tín chỉ, điểm trung bình của các SV trong môn học, tỉ lệ SV qua môn trong các nhóm (tính %). Kết quả được sắp xếp theo tỉ lệ SV qua môn học đó từ cao đến thấp. NV click vào 1 dòng của 1 MH thì hiện lên chi tiết bảng điểm của tất cả các SV đã học MH.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 13**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý đặt tour du lịch, được mô tả như sau:

- Mỗi tour (Mã tour, tên, nơi xuất phát, nơi đến, mô tả) có thể xuất phát vào nhiều ngày khác nhau, tùy vào ngày xuất phát và số lượng người mua tour cho mỗi đoàn sẽ có giá khác nhau.
- Mỗi khách hàng (Mã, tên, số ID, loại thẻ ID, số ĐT, email, địa chỉ) có thể mua vé nhiều tour khác nhau. Mỗi tour có thể mua số lượng vé khác nhau. Mỗi lần mua có xuất hóa đơn ghi rõ thông tin tour, ngày xuất phát, giá tour, số lượng khách, tên khách hàng đại diện, tổng số tiền thanh toán.
- Cùng một khách hàng có thể đi cùng một tour nhiều lần, chỉ khác nhau ở ngày xuất phát và giá vé.
- Khách hàng có thể trả vé, nếu trả trước giờ xuất phát trước 7 ngày thì phạt 10%, trước 5 ngày phạt 20%, trước 3 ngày phạt 50%, trước ít hơn 3 ngày phạt 100% giá ghi trên vé.

Anh/chị hãy thực hiện modul "**Quản lý thông tin về tour**" cho phép quản lý (QL) thực hiện thêm, sửa, xóa thông tin tour với mô tả chi tiết nghiệp vụ: QL chọn menu quản lý tour → trang quản lý hiện ra → QL chọn chức năng sửa thông tin tour → giao diện tìm tour theo tên hiện ra → QL nhập tên tour và click tìm kiếm → danh sách các tour có tên chứa từ khóa hiện ra → QL chọn sửa một tour → giao diện sửa tour hiện ra với các thông tin của tour đã chọn → QL nhập một số thông tin thay đổi và click cập nhật → hệ thống lưu thông tin vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 14**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý đặt tour du lịch, được mô tả như sau:

- Mỗi tour (Mã tour, tên, nơi xuất phát, nơi đến, mô tả) có thể xuất phát vào nhiều ngày khác nhau, tùy vào ngày xuất phát và số lượng người mua tour cho mỗi đoàn sẽ có giá khác nhau.
- Mỗi khách hàng (Mã, tên, số ID, loại thẻ ID, số ĐT, email, địa chỉ) có thể mua vé nhiều tour khác nhau. Mỗi tour có thể mua số lượng vé khác nhau. Mỗi lần mua có xuất hóa đơn ghi rõ thông tin tour, ngày xuất phát, giá tour, số lượng khách, tên khách hàng đại diện, tổng số tiền thanh toán.
- Cùng một khách hàng có thể đi cùng một tour nhiều lần, chỉ khác nhau ở ngày xuất phát và giá vé.
- Khách hàng có thể trả vé, nếu trả trước giờ xuất phát trước 7 ngày thì phạt 10%, trước 5 ngày phạt 20%, trước 3 ngày phạt 50%, trước ít hơn 3 ngày phạt 100% giá ghi trên vé.

Anh/chị hãy thực hiện modul "**Mua vé**" với các bước sau đây: Nhân viên chọn chức năng mua vé theo yêu cầu của khách → giao diện tìm tour (theo tên nơi đến) → NV nhập tên nơi đến và bấm tìm → kết quả hiện ra gồm danh sách các tour còn chỗ trống tương ứng với tiêu chí đã chọn, mỗi tour hiển thị đầy đủ thông tin + ngày xuất phát + giá tương ứng tại thời điểm tìm → NV chọn 1 tour theo lựa chọn của KH → hóa đơn (vé) hiện ra chi tiết: tên tour, nơi đi, nơi đến, ngày đi, tên khách đại diện đoàn, số ID, kiểu ID, địa chỉ khách, số điện thoại, email, số lượng khách, giá vé → NV chọn thanh toán → khách hàng thanh toán → hệ thống lưu kết quả vào và in vé cho khách hàng.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 15**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý đặt tour du lịch, được mô tả như sau:

- Mỗi tour (Mã tour, tên, nơi xuất phát, nơi đến, mô tả) có thể xuất phát vào nhiều ngày khác nhau, tùy vào ngày xuất phát và số lượng người mua tour cho mỗi đoàn sẽ có giá khác nhau.
- Mỗi khách hàng (Mã, tên, số ID, loại thẻ ID, số ĐT, email, địa chỉ) có thể mua vé nhiều tour khác nhau. Mỗi tour có thể mua số lượng vé khác nhau. Mỗi lần mua có xuất hóa đơn ghi rõ thông tin tour, ngày xuất phát, giá tour, số lượng khách, tên khách hàng đại diện, tổng số tiền thanh toán.
- Cùng một khách hàng có thể đi cùng một tour nhiều lần, chỉ khác nhau ở ngày xuất phát và giá vé.
- Khách hàng có thể trả vé, nếu trả trước giờ xuất phát trước 7 ngày thì phạt 10%, trước 5 ngày phạt 20%, trước 3 ngày phạt 50%, trước ít hơn 3 ngày phạt 100% giá ghi trên vé.

Anh/chị hãy thực hiện modul "**Khách hàng hủy bỏ đặt tour**" với các bước sau đây: Nhân viên chọn chức năng trả vé theo yêu cầu của khách → giao diện nhập mã vé hiện ra → NV nhập mã → kết quả hiện ra vé chi tiết: tên tour, nơi đi, nơi đến, ngày đi, tên khách đại diện đoàn, số ID, kiểu ID, địa chỉ khách, số điện thoại, email, số lượng khách, giá vé → NV chọn hủy vé → hệ thống hiện hóa đơn phạt bao gồm thông tin như trên vé + tiền phạt theo khung quy định → NV nhấn Ok → hệ thống lưu kết quả vào hệ thống, và nhân viên gửi lại phần tiền thừa cho khách hàng.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 16**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý đặt tour du lịch, được mô tả như sau:

- Mỗi tour (Mã tour, tên, nơi xuất phát, nơi đến, mô tả) có thể xuất phát vào nhiều ngày khác nhau, tùy vào ngày xuất phát và số lượng người mua tour cho mỗi đoàn sẽ có giá khác nhau.
- Mỗi khách hàng (Mã, tên, số ID, loại thẻ ID, số ĐT, email, địa chỉ) có thể mua vé nhiều tour khác nhau. Mỗi tour có thể mua số lượng vé khác nhau. Mỗi lần mua có xuất hóa đơn ghi rõ thông tin tour, ngày xuất phát, giá tour, số lượng khách, tên khách hàng đại diện, tổng số tiền thanh toán.
- Cùng một khách hàng có thể đi cùng một tour nhiều lần, chỉ khác nhau ở ngày xuất phát và giá vé.
- Khách hàng có thể trả vé, nếu trả trước giờ xuất phát trước 7 ngày thì phạt 10%, trước 5 ngày phạt 20%, trước 3 ngày phạt 50%, trước ít hơn 3 ngày phạt 100% giá ghi trên vé.

Anh/chị hãy thực hiện modul "**Thống kê tour theo doanh thu**" với các bước sau đây: Quản lý chọn chức năng thống kê các tour theo doanh thu → giao diện chọn thời gian thống kê (ngày bắt đầu - kết thúc) hiện ra → quản lý chọn xong bấm thống kê → kết quả hiện ra gồm danh sách các tour chi tiết: mã, tên, tên, nơi xuất phát, nơi đến, trung bình số khách/tour, tổng doanh thu. Sắp xếp theo tổng doanh thu, xếp từ cao đến thấp. NV click vào một dòng của một tour, hệ thống hiện ra danh sách chi tiết các hóa đơn của khách đã đặt mua tour đó, mỗi hóa đơn trên 1 dòng: id, tên khách, ngày giờ xuất phát, tổng số khách, tổng số tiền.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 17**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý đặt tour du lịch, được mô tả như sau:

- Mỗi tour (Mã tour, tên, nơi xuất phát, nơi đến, mô tả) có thể xuất phát vào nhiều ngày khác nhau, tùy vào ngày xuất phát và số lượng người mua tour cho mỗi đoàn sẽ có giá khác nhau.
- Mỗi khách hàng (Mã, tên, số ID, loại thẻ ID, số ĐT, email, địa chỉ) có thể mua vé nhiều tour khác nhau. Mỗi tour có thể mua số lượng vé khác nhau. Mỗi lần mua có xuất hóa đơn ghi rõ thông tin tour, ngày xuất phát, giá tour, số lượng khách, tên khách hàng đại diện, tổng số tiền thanh toán.
- Cùng một khách hàng có thể đi cùng một tour nhiều lần, chỉ khác nhau ở ngày xuất phát và giá vé.
- Khách hàng có thể trả vé, nếu trả trước giờ xuất phát trước 7 ngày thì phạt 10%, trước 5 ngày phạt 20%, trước 3 ngày phạt 50%, trước ít hơn 3 ngày phạt 100% giá ghi trên vé.

Anh/chị hãy thực hiện modul "**Thống kê doanh thu theo địa điểm**" với các bước sau đây: Quản lý chọn chức năng thống kê doanh thu theo địa điểm du lịch → giao diện chọn thời gian thống kê (ngày bắt đầu - kết thúc) hiện ra → quản lý chọn xong bấm thống kê → kết quả hiện ra gồm danh sách các địa điểm chi tiết: tên, số lượng tour đến địa điểm đấy, tổng số lượng khách đến địa điểm đấy, tổng doanh thu. Sắp xếp theo tổng doanh thu, xếp từ cao đến thấp. NV click vào một dòng của một địa điểm, hệ thống hiện ra danh sách chi tiết các hóa đơn của khách đã đặt mua tour qua địa điểm đó, mỗi hóa đơn trên 1 dòng: id, tên khách, ngày giờ xuất phát, tên tour, tổng số khách, tổng số tiền.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 18**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý việc gọi món trong một nhà hàng, được mô tả như sau:

- Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.
- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

Anh/chị hãy thực hiện modul "**Quản lý thông tin về món ăn**" cho phép quản lý (QL) thực hiện thêm, sửa, xóa thông tin món ăn với mô tả chi tiết nghiệp vụ: QL chọn menu quản lý món ăn → trang quản lý hiện ra → QL chọn chức năng sửa thông tin món ăn → giao diện tìm món ăn theo tên hiện ra → QL nhập tên món ăn và click tìm kiếm → danh sách các món ăn có tên chứa từ khóa hiện ra → QL chọn sửa một món ăn → giao diện sửa món ăn hiện ra với các thông tin của món ăn đã chọn → QL nhập một số thông tin thay đổi và click cập nhật → hệ thống lưu thông tin vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này



**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 19**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý việc gọi món trong một nhà hàng, được mô tả như sau:

- Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.
- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

Anh/chị hãy thực hiện modul "**Gọi món**" với các bước sau đây: Nhân viên chọn chức năng gọi món → giao diện bàn hiện ra với danh sách bàn và số hiệu sổ xuống → NV chọn bàn đúng với KH đang gọi món → Giao diện nhập món được gọi hiện ra → NV hỏi KH và nhập vào tên món ăn + chọn tìm → kết quả hiện ra gồm danh sách các món ăn chi tiết: mã, loại, tên, giá. → NV chọn 1 món ăn đúng như KH gọi và NV click chọn → Yêu cầu nhập số lượng → NV nhập số lượng và click OK → Tên món ăn + số lượng + số tiền tạm tính được thêm vào danh sách các món ăn đã chọn phía dưới. NV lặp lại các bước chọn món ăn này cho đến khi nhập vào được hết các món mà khách hàng trong bàn đã gọi. NV đọc lại để xác nhận với KH → NV click xác nhận → hệ thống lưu lại.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 20**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý việc gọi món trong một nhà hàng, được mô tả như sau:

- Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.
- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

Anh/chị hãy thực hiện modul "**Đặt bàn**" với các bước sau đây: Nhân viên chọn chức năng đặt bàn khi khách hàng gọi đến → giao diện tìm bàn trống hiện ra → NV nhập ngày + giờ đặt + số lượng khách và bấm tìm → kết quả hiện ra gồm danh sách các bàn còn trống vào ngày giờ đấy: mã, tên, số lượng khách tối đa, mô tả → NV chọn 1 bàn theo yêu cầu của KH → Giao diện nhập thông tin KH hiện ra → NV hỏi khách hàng và nhập mã, tên, số ĐT, email, địa chỉ và click tìm → Hệ thống hiện danh sách các khách hàng có cùng tên vừa nhập, mỗi khách hàng trên 1 dòng: mã, tên, số ĐT, email, địa chỉ → NV click vào dòng đúng với KH đag đặt (nếu không có thì lick thêm KH mới) → Hệ thống hiện lên giao diện xác nhận có đầy đủ thông tin bàn + thông tin KH + ngày giờ đặt → NV xác nhận với KH và click xác nhận → Hệ thống lưu thông tin vào CSDL.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 21**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý việc gọi món trong một nhà hàng, được mô tả như sau:

- Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.
- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

Anh/chị hãy thực hiện modul "**Lên menu sẵn dạng combo**" cho phép quản lý (QL) thực hiện thêm, sửa, xóa thông tin combo sẵn các món ăn với mô tả chi tiết nghiệp vụ: QL chọn menu quản lý combo → trang quản lý hiện ra → QL chọn chức năng thêm combo → giao diện thêm combo hiện ra với các ô nhập tên combo và nút thêm món ăn vào combo → QL click thêm món ăn vào combo → giao diện tìm món ăn theo tên hiện ra → QL nhập tên món ăn và click tìm kiếm → danh sách các món ăn có tên chứa từ khóa hiện ra → QL chọn một món ăn → hệ thống quay về giao diện thêm combo với món ăn vừa chọn được thêm vào combo → QL lặp lại cho đến khi thêm xong các món ăn cần cho vào combo và sau đó QL click cập nhật → hệ thống lưu thông tin vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 22**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý việc gọi món trong một nhà hàng, được mô tả như sau:

- Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.
- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

Anh/chị hãy thực hiện modul "**Thanh toán**" với các bước sau đây: KH yêu cầu VN thanh toán → Nhân viên chọn chức năng thanh toán → giao diện chọn bàn hiện ra với danh sách bàn và số hiệu sổ xuống → NV chọn bàn đúng với bàn của KH → Giao diện hóa đơn chi tiết của bàn hiện ra như mô tả ở trên → NV hỏi KH có phiếu giảm giá không → nếu có thì click thêm phiếu giảm giá + nhập mã → giao diện hóa đơn thêm dòng phiếu giảm giá và cập nhật lại tổng tiền phải thanh toán → NV báo KH số tiền → Sau khi thanh toán, NV click xác nhận → hệ thống lưu lại và in hóa đơn chi tiết cho KH.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 23**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý việc gọi món trong một nhà hàng, được mô tả như sau:

- Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.
- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

Anh/chị hãy thực hiện modul "***Thống kê lượng khách theo khung giờ***" với các bước sau đây: Quản lý chọn chức năng thống kê lượng khách theo khung giờ → giao diện chọn thời gian thống kê (ngày bắt đầu - kết thúc) hiện ra → quản lý chọn xong bấm thống kê → kết quả hiện ra gồm danh sách các khung giờ chi tiết: khung giờ từ mấy giờ đến mấy giờ trong ngày, trung bình số lượng khách, trung bình doanh thu/đầu khách, tổng doanh thu của khung giờ. Sắp xếp theo tổng doanh thu, xếp từ cao đến thấp. NV click vào một khung giờ, hệ thống hiện lên chi tiết các hóa đơn của khách đã dùng trong khung giờ đấy, mỗi hóa đơn trên 1 dòng: mã, tên khách, ngày, tổng số món gọi, tổng số tiền thanh toán.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 24**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý việc gọi món trong một nhà hàng, được mô tả như sau:

- Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.
- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

Anh/chị hãy thực hiện modul "***Thống kê doanh thu theo tháng***" với các bước sau đây: QL chọn menu thống kê → chọn thống kê doanh thu theo món ăn → nhập thời gian bắt đầu và kết thúc thống kê → danh sách các món ăn có doanh thu lớn nhất được hiện ra, mỗi dòng cho 1 món ăn: Mã, tên, tổng số lượng khách đã dùng, tổng doanh thu thu được, sắp xếp theo chiều giảm dần tổng doanh thu. NV click vào 1 dòng của 1 tháng, hệ thống hiện lên chi tiết các hóa đơn của khách trong tháng, mỗi hóa đơn trên dòng: id, tên khách, ngày giờ, tổng số món gọi, tổng số tiền thanh toán.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 25**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm quản lý việc gọi món trong một nhà hàng, được mô tả như sau:

- Nhà hàng có nhiều bàn (Mã bàn, tên, số lượng khách tối đa, mô tả). Nhiều bàn nhỏ có thể gộp lại thành một bàn lớn khi có yêu cầu từ đoàn khách có số lượng lớn.
- Mỗi bàn, có thể bị đặt nhiều lần khác nhau trong ngày, hoặc khác ngày.
- Mỗi khách hàng (Mã, tên, số ĐT, email, địa chỉ) có thể đặt bàn nhiều lần, mỗi lần có thể đặt nhiều bàn (trường hợp này sẽ bị gộp thành đặt 1 bàn)
- Nhà hàng có thể lên combo dạng kết hợp sẵn một số món ăn đủ cho 1 bữa ăn cho một người ăn. Khách hàng có thể gọi combo có sẵn như thế này.
- Khách hàng ở mỗi bàn có thể gọi nhiều món ăn (Mã, loại, tên, mô tả, giá hiện tại) hoặc combo. Mỗi món ăn (combo) có thể bị gọi với số lượng khác nhau.
- Khi thanh toán, hóa đơn ghi đầy đủ thông tin: mã bàn, tên và mã nhân viên thanh toán, tên khách hàng nếu có, sau đó là một bảng, mỗi dòng chứa thông tin một món (combo) đã dùng: id, tên, đơn giá, số lượng, thành tiền. Dòng cuối cùng ghi tổng số tiền của hóa đơn.

Anh/chị hãy thực hiện modul "**Thống kê món ăn bán chạy**" với các bước sau đây: Quản lý chọn chức năng thống kê món ăn bán chạy → giao diện chọn thời gian thống kê (ngày bắt đầu - kết thúc) hiện ra → quản lý chọn xong bấm thống kê → kết quả hiện ra gồm danh sách các món ăn chi tiết: mã, loại, tên, tổng số lượt bán, tổng doanh thu. Sắp xếp theo tổng doanh thu, xếp từ cao đến thấp. NV click vào 1 dòng của 1 món ăn, hệ thống hiện lên chi tiết danh sách các lần món ăn được gọi: id, tên khách, ngày giờ, số lượng, thành tiền.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 26**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm hỗ trợ quản lý kho vật tư, được mô tả như sau:

- Mỗi hàng hóa (Mã hàng, tên, mô tả) có thể được nhập nhiều lần khác nhau, mỗi lần nhập có số lượng khác nhau và giá nhập khác nhau, đến từ một nhà cung cấp (mã NCC, tên NCC, địa chỉ, số ĐT) khác nhau
- Mỗi lần nhập hàng có thể nhập nhiều hàng hóa khác nhau
- Mỗi lần nhập có một phiếu nhập ghi thông tin nhà cung cấp, tiếp theo là danh sách các mặt hàng nhập vào, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn nhập
- Tương tự, mỗi hàng hóa có thể xuất đi nhiều lần khác nhau, mỗi lần cho các đại lý con (mã ĐL, tên ĐL, địa chỉ, số ĐT) khác nhau, với số lượng khác nhau và giá xuất khác nhau
- Mỗi lần xuất có thể xuất nhiều hàng khác nhau, miễn sao số lượng xuất không vượt quá số lượng hàng còn trong kho
- Mỗi lần xuất có một phiếu xuất ghi thông tin đại lý con, tiếp theo là danh sách các mặt hàng xuất đi, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn xuất.

Anh/chị hãy thực hiện modul "**Quản lý thông tin vật tư**" cho phép quản lý (QL) thực hiện thêm, sửa, xóa thông tin mặt hàng với mô tả chi tiết nghiệp vụ: QL chọn menu quản lý mặt hàng → trang quản lý hiện ra → QL chọn chức năng sửa thông tin mặt hàng → giao diện tìm mặt hàng theo tên hiện ra → QL nhập tên mặt hàng và click tìm kiếm → danh sách các mặt hàng có tên chứa từ khóa hiện ra → QL chọn sửa một mặt hàng → giao diện sửa mặt hàng hiện ra với các thông tin của mặt hàng đã chọn → QL nhập một số thông tin thay đổi và click cập nhật → hệ thống lưu thông tin vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này



**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 27**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm hỗ trợ quản lý kho vật tư, được mô tả như sau:

- Mỗi hàng hóa (Mã hàng, tên, mô tả) có thể được nhập nhiều lần khác nhau, mỗi lần nhập có số lượng khác nhau và giá nhập khác nhau, đến từ một nhà cung cấp (mã NCC, tên NCC, địa chỉ, số ĐT) khác nhau
- Mỗi lần nhập hàng có thể nhập nhiều hàng hóa khác nhau
- Mỗi lần nhập có một phiếu nhập ghi thông tin nhà cung cấp, tiếp theo là danh sách các mặt hàng nhập vào, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn nhập
- Tương tự, mỗi hàng hóa có thể xuất đi nhiều lần khác nhau, mỗi lần cho các đại lý con (mã ĐL, tên ĐL, địa chỉ, số ĐT) khác nhau, với số lượng khác nhau và giá xuất khác nhau
- Mỗi lần xuất có thể xuất nhiều hàng khác nhau, miễn sao số lượng xuất không vượt quá số lượng hàng còn trong kho
- Mỗi lần xuất có một phiếu xuất ghi thông tin đại lý con, tiếp theo là danh sách các mặt hàng xuất đi, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn xuất.

Anh/chị hãy thực hiện modul "***lập phiếu xuất hàng***" với các bước sau đây: Nhân viên chọn menu xuất hàng → trang xuất hàng hiện ra với ô tìm kiếm đại lý con (ĐLC) → NV nhập tên ĐL và click tìm → hệ thống hiện lên danh sách các ĐL có tên chứa tên vừa nhập → NV click chọn dòng của ĐL đúng với ĐL nhập (trường hợp ĐL mới thì phải thêm mới vào) → hệ thống hiện lên giao diện tìm hàng xuất → NV nhập tên hàng và click tìm → hệ thống hiện lên danh sách các MH có tên chứa từ khóa vừa nhập → nhân viên chọn tên hàng trong danh sách hàng hóa có sẵn + nhập số lượng + đơn giá → MH xuất hiện vào danh sách MH xuất trong hóa đơn → lặp đến khi hết các hàng cần xuất vào thì submit → báo xuất thành công và in ra hóa đơn xuất như đã mô tả.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 28**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm hỗ trợ quản lí kho vật tư, được mô tả như sau:

- Mỗi hàng hóa (Mã hàng, tên, mô tả) có thể được nhập nhiều lần khác nhau, mỗi lần nhập có số lượng khác nhau và giá nhập khác nhau, đến từ một nhà cung cấp (mã NCC, tên NCC, địa chỉ, số ĐT) khác nhau
- Mỗi lần nhập hàng có thể nhập nhiều hàng hóa khác nhau
- Mỗi lần nhập có một phiếu nhập ghi thông tin nhà cung cấp, tiếp theo là danh sách các mặt hàng nhập vào, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn nhập
- Tương tự, mỗi hàng hóa có thể xuất đi nhiều lần khác nhau, mỗi lần cho các đại lí con (mã ĐL, tên ĐL, địa chỉ, số ĐT) khác nhau, với số lượng khác nhau và giá xuất khác nhau
- Mỗi lần xuất có thể xuất nhiều hàng khác nhau, miễn sao số lượng xuất không vượt quá số lượng hàng còn trong kho
- Mỗi lần xuất có một phiếu xuất ghi thông tin đại lí con, tiếp theo là danh sách các mặt hàng xuất đi, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn xuất.

Anh/chị hãy thực hiện modul "**Lập phiếu nhập hàng**" với các bước sau đây: Nhân viên chọn menu nhập hàng → trang nhập hàng hiện ra với ô tìm NCC theo tên → NV nhập tên + click tìm → hệ thống hiện lên danh sách các NCC chứa tên vừa nhập vào → NV click vào NCC đang nhập (nếu NCC mới thì thêm mới) → Lập các bước sau cho hết hàng nhập: NV click chọn tìm MH theo tên → nhập tên + click tìm → hệ thống hiện lên danh sách các MH chứa tên vừa nhập → nhân viên chọn tên hàng trong danh sách hàng hóa có sẵn (nếu hàng mới thì chọn nhập mới) + nhập số lượng → MH đó sẽ được thêm vào danh sách các MH nhập của hóa đơn → lặp đến khi hết các hàng nhập vào thì submit → báo nhập thành công và in ra hóa đơn nhập như đã mô tả.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 29**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm hỗ trợ quản lý kho vật tư, được mô tả như sau:

- Mỗi hàng hóa (Mã hàng, tên, mô tả) có thể được nhập nhiều lần khác nhau, mỗi lần nhập có số lượng khác nhau và giá nhập khác nhau, đến từ một nhà cung cấp (mã NCC, tên NCC, địa chỉ, số ĐT) khác nhau
- Mỗi lần nhập hàng có thể nhập nhiều hàng hóa khác nhau
- Mỗi lần nhập có một phiếu nhập ghi thông tin nhà cung cấp, tiếp theo là danh sách các mặt hàng nhập vào, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn nhập
- Tương tự, mỗi hàng hóa có thể xuất đi nhiều lần khác nhau, mỗi lần cho các đại lý con (mã ĐL, tên ĐL, địa chỉ, số ĐT) khác nhau, với số lượng khác nhau và giá xuất khác nhau
- Mỗi lần xuất có thể xuất nhiều hàng khác nhau, miễn sao số lượng xuất không vượt quá số lượng hàng còn trong kho
- Mỗi lần xuất có một phiếu xuất ghi thông tin đại lý con, tiếp theo là danh sách các mặt hàng xuất đi, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn xuất.

Anh/chị hãy thực hiện modul "***Thống kê sản phẩm bán chạy***" với các bước sau đây: Nhân viên chọn menu thống kê → chọn chức năng thống kê sản phẩm bán chạy → nhập khoảng thời gian thống kê (bắt đầu - kết thúc) → kết quả hiện ra danh sách các sản phẩm theo thứ tự bán được tổng số lượng nhiều nhất đến ít dần trong khoảng thời gian đã chọn, mỗi dòng có các thông tin: mã hàng, tên hàng, số lượng đã bán được, tổng số tiền đã thu được từ sản phẩm ấy trong khoảng thời gian đã chọn. NV click vào một dòng của 1 sản phẩm thì hiện lên thống kê chi tiết các hóa đơn của các đại lý con đã mua sản phẩm đấy.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 30**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu chúng ta phát triển một phần mềm hỗ trợ quản lý kho vật tư, được mô tả như sau:

- Mỗi hàng hóa (Mã hàng, tên, mô tả) có thể được nhập nhiều lần khác nhau, mỗi lần nhập có số lượng khác nhau và giá nhập khác nhau, đến từ một nhà cung cấp (mã NCC, tên NCC, địa chỉ, số ĐT) khác nhau
- Mỗi lần nhập hàng có thể nhập nhiều hàng hóa khác nhau
- Mỗi lần nhập có một phiếu nhập ghi thông tin nhà cung cấp, tiếp theo là danh sách các mặt hàng nhập vào, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn nhập
- Tương tự, mỗi hàng hóa có thể xuất đi nhiều lần khác nhau, mỗi lần cho các đại lý con (mã ĐL, tên ĐL, địa chỉ, số ĐT) khác nhau, với số lượng khác nhau và giá xuất khác nhau
- Mỗi lần xuất có thể xuất nhiều hàng khác nhau, miễn sao số lượng xuất không vượt quá số lượng hàng còn trong kho
- Mỗi lần xuất có một phiếu xuất ghi thông tin đại lý con, tiếp theo là danh sách các mặt hàng xuất đi, mỗi mặt hàng có đầy đủ thông tin: mã hàng, tên hàng, số lượng, đơn giá, thành tiền (tự động tính) và dòng cuối cùng là tổng tiền của hóa đơn xuất.

Anh/chị hãy thực hiện modul "**Thống kê đại lý tiêu thụ mạnh**" với các bước sau đây: Nhân viên chọn menu thống kê → chọn chức năng thống kê đại lý tiêu thụ hàng đầu → nhập khoảng thời gian thống kê (bắt đầu - kết thúc) → kết quả hiện ra danh sách các đại lý tiêu thụ theo thứ tự bán được tổng doanh thu nhiều nhất đến ít dần trong khoảng thời gian đã chọn, mỗi dòng có các thông tin: mã đại lý, tên đại lý, tổng số tiền đã thu được từ đại lý ấy trong khoảng thời gian đã chọn. NV click vào 1 dòng của đại lý thì hiện lên chi tiết danh sách các hóa đơn (ngày, tổng số hàng, tổng số tiền) của mỗi lần đại lý con đấy đã nhập hàng.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 31**  
**Thời gian làm bài: 60 phút**

Liên đoàn cờ vua thế giới (FIDE) yêu cầu anh/chị phát triển một phần mềm quản lý Giải đấu vô địch thế giới với mô tả như sau:

- Mỗi giải đấu (Mã, tên, năm, lần tổ chức, địa điểm, mô tả) cho phép nhiều cờ thủ (mã, tên, năm sinh, quốc tịch, hệ số Elo, ghi chú) tham gia.
- Có thể có hàng trăm cờ thủ tham gia, nhưng mỗi cờ thủ phải thi đấu 11 trận theo hệ Thụy Sĩ
- Ở ván thứ nhất, các cờ thủ được xếp hạng theo thứ tự hệ số Elo từ cao đến thấp. Sau đó đi từ trên xuống dưới bảng sắp xếp, hai cờ thủ đứng kề nhau sẽ tạo thành một cặp đấu cho vòng 1.
- Ở mỗi vòng đấu, thắng được 1 điểm, hòa được 0.5 điểm, thua được 0 điểm. Sau mỗi vòng đấu, kết quả từng trận được cập nhật theo các cặp đấu đã lên lịch trước đó. Đồng thời hệ số Elo tăng hay giảm sau mỗi vòng đấu cũng được cập nhật (Tính theo công thức của FIDE, chỉ cần nhập kết quả vào).
- Bắt đầu từ ván thứ 2, bảng xếp hạng tạm thời sau vòng đấu trước đó được xếp theo thứ tự các tiêu chí: tổng điểm (giảm dần), tổng điểm của các đối thủ đã gặp (giảm dần), hệ số Elo (giảm dần). Và cặp đấu được xác định như sau, đi từ đầu đến cuối bảng xếp hạng tạm thời, với mỗi cờ thủ chưa có cặp, đối thủ cả cờ thủ đó là cờ thủ đầu tiên gặp phải và thỏa mãn: chưa có cặp, và chưa gặp cờ thủ đang xem xét.
- Sau 11 vòng đấu như vậy, cờ thủ đứng đầu bảng xếp hạng sẽ là nhà vô địch.

Anh/chị hãy thực hiện modul "***Quản lý thông tin kì thủ***" cho phép quản lý (QL) thực hiện thêm, sửa, xóa thông tin kì thủ với mô tả chi tiết nghiệp vụ: QL chọn menu quản lý kì thủ → trang quản lý hiện ra → QL chọn chức năng sửa thông tin kì thủ → giao diện tìm kì thủ theo tên hiện ra → QL nhập tên kì thủ và click tìm kiếm → danh sách các kì thủ có tên chứa từ khóa hiện ra → QL chọn sửa một kì thủ → giao diện sửa kì thủ hiện ra với các thông tin của kì thủ đã chọn → QL nhập một số thông tin thay đổi và click cập nhật → hệ thống lưu thông tin vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 32**  
**Thời gian làm bài: 60 phút**

Liên đoàn cờ vua thế giới (FIDE) yêu cầu anh/chị phát triển một phần mềm quản lý Giải đấu vô địch thế giới với mô tả như sau:

- Mỗi giải đấu (Mã, tên, năm, lần tổ chức, địa điểm, mô tả) cho phép nhiều cờ thủ (mã, tên, năm sinh, quốc tịch, hệ số Elo, ghi chú) tham gia.
- Có thể có hàng trăm cờ thủ tham gia, nhưng mỗi cờ thủ phải thi đấu 11 trận theo hệ Thụy Sĩ
- Ở ván thứ nhất, các cờ thủ được xếp hạng theo thứ tự hệ số Elo từ cao đến thấp. Sau đó đi từ trên xuống dưới bảng sắp xếp, hai cờ thủ đứng kề nhau sẽ tạo thành một cặp đấu cho vòng 1.
- Ở mỗi vòng đấu, thắng được 1 điểm, hòa được 0.5 điểm, thua được 0 điểm. Sau mỗi vòng đấu, kết quả từng trận được cập nhật theo các cặp đấu đã lên lịch trước đó. Đồng thời hệ số Elo tăng hay giảm sau mỗi vòng đấu cũng được cập nhật (Tính theo công thức của FIDE, chỉ cần nhập kết quả vào).
- Bắt đầu từ ván thứ 2, bảng xếp hạng tạm thời sau vòng đấu trước đó được xếp theo thứ tự các tiêu chí: tổng điểm (giảm dần), tổng điểm của các đối thủ đã gặp (giảm dần), hệ số Elo (giảm dần). Và cặp đấu được xác định như sau, đi từ đầu đến cuối bảng xếp hạng tạm thời, với mỗi cờ thủ chưa có cặp, đối thủ cả cờ thủ đó là cờ thủ đầu tiên gặp phải và thỏa mãn: chưa có cặp, và chưa gặp cờ thủ đang xem xét.
- Sau 11 vòng đấu như vậy, cờ thủ đứng đầu bảng xếp hạng sẽ là nhà vô địch.

Anh/chị hãy thực hiện modul "**Cập nhật kết quả**" với các bước sau đây: Ban tổ chức (BTC) chọn menu cập nhật kết quả → trang cập nhật kết quả hiện ra → BTC chọn vòng đấu từ danh sách sổ ra + chọn cặp đấu từ danh sách sổ ra theo vòng đấu + nhập số điểm và điểm Elo cho 2 cờ thủ của trận đấu + click Cập nhật → Hệ thống thông báo lưu thành công kết quả trận đấu và quay về trang chọn vòng đấu + trận đấu.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 33**  
**Thời gian làm bài: 60 phút**

Liên đoàn cờ vua thế giới (FIDE) yêu cầu anh/chị phát triển một phần mềm quản lý Giải đấu vô địch thế giới với mô tả như sau:

- Mỗi giải đấu (Mã, tên, năm, lần tổ chức, địa điểm, mô tả) cho phép nhiều cờ thủ (mã, tên, năm sinh, quốc tịch, hệ số Elo, ghi chú) tham gia.
- Có thể có hàng trăm cờ thủ tham gia, nhưng mỗi cờ thủ phải thi đấu 11 trận theo hệ Thụy Sĩ
- Ở ván thứ nhất, các cờ thủ được xếp hạng theo thứ tự hệ số Elo từ cao đến thấp. Sau đó đi từ trên xuống dưới bảng sắp xếp, hai cờ thủ đứng kề nhau sẽ tạo thành một cặp đấu cho vòng 1.
- Ở mỗi vòng đấu, thắng được 1 điểm, hòa được 0.5 điểm, thua được 0 điểm. Sau mỗi vòng đấu, kết quả từng trận được cập nhật theo các cặp đấu đã lên lịch trước đó. Đồng thời hệ số Elo tăng hay giảm sau mỗi vòng đấu cũng được cập nhật (Tính theo công thức của FIDE, chỉ cần nhập kết quả vào).
- Bắt đầu từ ván thứ 2, bảng xếp hạng tạm thời sau vòng đấu trước đó được xếp theo thứ tự các tiêu chí: tổng điểm (giảm dần), tổng điểm của các đối thủ đã gặp (giảm dần), hệ số Elo (giảm dần). Và cặp đấu được xác định như sau, đi từ đầu đến cuối bảng xếp hạng tạm thời, với mỗi cờ thủ chưa có cặp, đối thủ cả cờ thủ đó là cờ thủ đầu tiên gặp phải và thỏa mãn: chưa có cặp, và chưa gặp cờ thủ đang xem xét.
- Sau 11 vòng đấu như vậy, cờ thủ đứng đầu bảng xếp hạng sẽ là nhà vô địch.

Anh/chị hãy thực hiện modul "**Xem bảng xếp hạng**" với các bước sau đây: Ban tổ chức (BTC) chọn menu thông kê → chọn chức năng xem bảng xếp hạng sau từng vòng đấu → chọn vòng đấu trong danh sách sỏ ra → kết quả hiện ra danh sách các cờ thủ, mỗi người có đầy đủ thông tin: id, tên, năm sinh, quốc tịch, tổng điểm, tổng điểm đối thủ đã gặp, hệ số Elo tức thời. Sắp xếp theo thứ tự đã mô tả ở trên.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 34**  
**Thời gian làm bài: 60 phút**

Liên đoàn cờ vua thế giới (FIDE) yêu cầu anh/chị phát triển một phần mềm quản lý Giải đấu vô địch thế giới với mô tả như sau:

- Mỗi giải đấu (Mã, tên, năm, lần tổ chức, địa điểm, mô tả) cho phép nhiều cờ thủ (mã, tên, năm sinh, quốc tịch, hệ số Elo, ghi chú) tham gia.
- Có thể có hàng trăm cờ thủ tham gia, nhưng mỗi cờ thủ phải thi đấu 11 trận theo hệ Thụy Sĩ
- Ở ván thứ nhất, các cờ thủ được xếp hạng theo thứ tự hệ số Elo từ cao đến thấp. Sau đó đi từ trên xuống dưới bảng sắp xếp, hai cờ thủ đứng kề nhau sẽ tạo thành một cặp đấu cho vòng 1.
- Ở mỗi vòng đấu, thắng được 1 điểm, hòa được 0.5 điểm, thua được 0 điểm. Sau mỗi vòng đấu, kết quả từng trận được cập nhật theo các cặp đấu đã lên lịch trước đó. Đồng thời hệ số Elo tăng hay giảm sau mỗi vòng đấu cũng được cập nhật (Tính theo công thức của FIDE, chỉ cần nhập kết quả vào).
- Bắt đầu từ ván thứ 2, bảng xếp hạng tạm thời sau vòng đấu trước đó được xếp theo thứ tự các tiêu chí: tổng điểm (giảm dần), tổng điểm của các đối thủ đã gặp (giảm dần), hệ số Elo (giảm dần). Và cặp đấu được xác định như sau, đi từ đầu đến cuối bảng xếp hạng tạm thời, với mỗi cờ thủ chưa có cặp, đối thủ cả cờ thủ đó là cờ thủ đầu tiên gặp phải và thỏa mãn: chưa có cặp, và chưa gặp cờ thủ đang xem xét.
- Sau 11 vòng đấu như vậy, cờ thủ đứng đầu bảng xếp hạng sẽ là nhà vô địch.

Anh/chị hãy thực hiện modul "**Xếp cặp thi đấu**" với các bước sau đây: an tổ chức (BTC) chọn menu xếp cặp thi đấu → trang xếp cặp thi đấu hiện ra → BTC chọn vòng đấu trước đó trong danh sách sổ xuống → hệ thống hiện bảng xếp hạng hiện tại sau vòng đấu trước đó + nút Xếp lịch → BTC click nút Xếp lịch → Hệ thống tự động xếp cặp cho các cờ thủ theo luật mô tả ở trên, và hiện danh sách các bàn đấu theo đúng thứ tự các cặp đấu → BTC click Lưu → Hệ thống lưu lịch thi đấu của vòng mới vào CSDL

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này



**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 35**  
**Thời gian làm bài: 60 phút**

Liên đoàn cờ vua thế giới (FIDE) yêu cầu anh/chị phát triển một phần mềm quản lý Giải đấu vô địch thế giới với mô tả như sau:

- Mỗi giải đấu (Mã, tên, năm, lần tổ chức, địa điểm, mô tả) cho phép nhiều cờ thủ (mã, tên, năm sinh, quốc tịch, hệ số Elo, ghi chú) tham gia.
- Có thể có hàng trăm cờ thủ tham gia, nhưng mỗi cờ thủ phải thi đấu 11 trận theo hệ Thụy Sĩ
- Ở ván thứ nhất, các cờ thủ được xếp hạng theo thứ tự hệ số Elo từ cao đến thấp. Sau đó đi từ trên xuống dưới bảng sắp xếp, hai cờ thủ đứng kề nhau sẽ tạo thành một cặp đấu cho vòng 1.
- Ở mỗi vòng đấu, thắng được 1 điểm, hòa được 0.5 điểm, thua được 0 điểm. Sau mỗi vòng đấu, kết quả từng trận được cập nhật theo các cặp đấu đã lên lịch trước đó. Đồng thời hệ số Elo tăng hay giảm sau mỗi vòng đấu cũng được cập nhật (Tính theo công thức của FIDE, chỉ cần nhập kết quả vào).
- Bắt đầu từ ván thứ 2, bảng xếp hạng tạm thời sau vòng đấu trước đó được xếp theo thứ tự các tiêu chí: tổng điểm (giảm dần), tổng điểm của các đối thủ đã gặp (giảm dần), hệ số Elo (giảm dần). Và cặp đấu được xác định như sau, đi từ đầu đến cuối bảng xếp hạng tạm thời, với mỗi cờ thủ chưa có cặp, đối thủ cả cờ thủ đó là cờ thủ đầu tiên gặp phải và thỏa mãn: chưa có cặp, và chưa gặp cờ thủ đang xem xét.
- Sau 11 vòng đấu như vậy, cờ thủ đứng đầu bảng xếp hạng sẽ là nhà vô địch.

Anh/chị hãy thực hiện modul "**Thống kê thay đổi Elo**" với các bước sau đây: Ban tổ chức (BTC) chọn menu thống kê → chọn chức năng thống kê thay đổi Elo của các cờ thủ sau giải → kết quả hiện ra danh sách các cờ thủ, mỗi cờ thủ được hiện đầy đủ thông tin: mã, tên, năm sinh, quốc tịch, hệ số Elo cũ, hệ số Elo mới, hệ số Elo đã tăng/giảm. Sắp xếp theo thứ tự giảm dần của mức tăng giảm hệ số Elo của các kì thủ, tiếp đến là giảm dần của hệ số Elo mới, sau giải. Click vào 1 dòng của một cờ thủ → hệ thống hiện lên chi tiết các trận cờ thủ đấy đã đấu, mỗi trận trên 1 dòng: id, tên đối thủ, mức tăng giảm Elo.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 36**  
**Thời gian làm bài: 60 phút**

Ban tổ chức đua xe công thức 1 (F1) đặt hàng anh/chị phát triển một phần mềm quản lý kết quả giải đua với mô tả như sau:

- Mỗi năm có một giải. Một giải bao gồm nhiều chặng đua diễn ra trên khắp thế giới (Mã chặng, tên, số vòng đua, địa điểm, thời gian, mô tả).
- Mỗi giải có nhiều đội đua tham gia (Mã, tên, hãng, mô tả).
- Mỗi đội đua có nhiều tay đua (mã, tên, ngày sinh, quốc tịch, tiểu sử). Nhưng ở mỗi chặng đua, mỗi đội chỉ được phép cho tối đa 2 tay đua tham dự.
- Mỗi chặng đua, kết quả xếp theo thứ tự về đích (thời gian) và điểm số chỉ được tính cho top 10 người về đích sớm nhất, lần lượt theo các thứ tự về đích là 25, 18, 15, 12, 10, 8, 6, 4, 2, 1.
- Nếu tay đua nằm trong top 10 nhưng không về đích do bỏ cuộc hoặc tai nạn thì 0 điểm.
- Điểm số và thời gian của từng tay đua sẽ được cộng dồn giữa các chặng để quyết định giải cá nhân và giải đồng đội của mùa giải.

Anh/chị hãy thực hiện modul "**Quản lý tay đua**" cho phép quản lý (QL) thực hiện thêm, sửa, xóa thông tin tay đua với mô tả chi tiết nghiệp vụ: QL chọn menu quản lý tay đua → trang quản lý hiện ra → QL chọn chức năng sửa thông tin tay đua → giao diện tìm tay đua theo tên hiện ra → QL nhập tên tay đua và click tìm kiếm → danh sách các tay đua có tên chứa từ khóa hiện ra → QL chọn sửa một tay đua → giao diện sửa tay đua hiện ra với các thông tin của tay đua đã chọn → QL nhập một số thông tin thay đổi và click cập nhật → hệ thống lưu thông tin vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 37**  
**Thời gian làm bài: 60 phút**

Ban tổ chức đua xe công thức 1 (F1) đặt hàng anh/chị phát triển một phần mềm quản lý kết quả giải đua với mô tả như sau:

- Mỗi năm có một giải. Một giải bao gồm nhiều chặng đua diễn ra trên khắp thế giới (Mã chặng, tên, số vòng đua, địa điểm, thời gian, mô tả).
- Mỗi giải có nhiều đội đua tham gia (Mã, tên, hãng, mô tả).
- Mỗi đội đua có nhiều tay đua (mã, tên, ngày sinh, quốc tịch, tiểu sử). Nhưng ở mỗi chặng đua, mỗi đội chỉ được phép cho tối đa 2 tay đua tham dự.
- Mỗi chặng đua, kết quả xếp theo thứ tự về đích (thời gian) và điểm số chỉ được tính cho top 10 người về đích sớm nhất, lần lượt theo các thứ tự về đích là 25, 18, 15, 12, 10, 8, 6, 4, 2, 1.
- Nếu tay đua nằm trong top 10 nhưng không về đích do bỏ cuộc hoặc tai nạn thì 0 điểm.
- Điểm số và thời gian của từng tay đua sẽ được cộng dồn giữa các chặng để quyết định giải cá nhân và giải đồng đội của mùa giải.

Anh/chị hãy thực hiện modul "**Đăng kí thi đấu**" với các bước sau đây: Ban tổ chức (BTC) chọn chức năng đăng kí tay đua → giao diện đăng kí tay đua cho mỗi chặng đấu hiện ra → BTC chọn chặng đua từ danh sách xổ xuống + chọn đội đua từ danh sách xổ xuống → danh sách các tay đua của đội đua đã chọn hiện ra, xếp theo abc của họ tên → BTC tích chọn đúng 2 tay đua theo yêu cầu của đội + click Đăng kí → Hệ thống lưu thông tin và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 38**  
**Thời gian làm bài: 60 phút**

Ban tổ chức đua xe công thức 1 (F1) đặt hàng anh/chị phát triển một phần mềm quản lý kết quả giải đua với mô tả như sau:

- Mỗi năm có một giải. Một giải bao gồm nhiều chặng đua diễn ra trên khắp thế giới (Mã chặng, tên, số vòng đua, địa điểm, thời gian, mô tả).
- Mỗi giải có nhiều đội đua tham gia (Mã, tên, hãng, mô tả).
- Mỗi đội đua có nhiều tay đua (mã, tên, ngày sinh, quốc tịch, tiểu sử). Nhưng ở mỗi chặng đua, mỗi đội chỉ được phép cho tối đa 2 tay đua tham dự.
- Mỗi chặng đua, kết quả xếp theo thứ tự về đích (thời gian) và điểm số chỉ được tính cho top 10 người về đích sớm nhất, lần lượt theo các thứ tự về đích là 25, 18, 15, 12, 10, 8, 6, 4, 2, 1.
- Nếu tay đua nằm trong top 10 nhưng không về đích do bỏ cuộc hoặc tai nạn thì 0 điểm.
- Điểm số và thời gian của từng tay đua sẽ được cộng dồn giữa các chặng để quyết định giải cá nhân và giải đồng đội của mùa giải.

Anh/chị hãy thực hiện modul "**Cập nhật kết quả**" với các bước sau đây: Ban tổ chức (BTC) chọn chức năng nhập kết quả chặng đua → giao diện nhập kết quả hiện ra → BTC chọn tên chặng đua từ danh sách xổ xuống → Danh sách các tay đua đã đăng ký thi đấu cho chặng đua hiện ra dưới dạng bảng, mỗi dòng chứa các ô trống nhập thời gian về đích, số vòng đua hoàn thành → BTC nhập đầy đủ kết quả tất cả các tay đua và click Lưu → Hệ thống lưu kết quả vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 39**  
**Thời gian làm bài: 60 phút**

Ban tổ chức đua xe công thức 1 (F1) đặt hàng anh/chị phát triển một phần mềm quản lý kết quả giải đua với mô tả như sau:

- Mỗi năm có một giải. Một giải bao gồm nhiều chặng đua diễn ra trên khắp thế giới (Mã chặng, tên, số vòng đua, địa điểm, thời gian, mô tả).
- Mỗi giải có nhiều đội đua tham gia (Mã, tên, hãng, mô tả).
- Mỗi đội đua có nhiều tay đua (mã, tên, ngày sinh, quốc tịch, tiểu sử). Nhưng ở mỗi chặng đua, mỗi đội chỉ được phép cho tối đa 2 tay đua tham dự.
- Mỗi chặng đua, kết quả xếp theo thứ tự về đích (thời gian) và điểm số chỉ được tính cho top 10 người về đích sớm nhất, lần lượt theo các thứ tự về đích là 25, 18, 15, 12, 10, 8, 6, 4, 2, 1.
- Nếu tay đua nằm trong top 10 nhưng không về đích do bỏ cuộc hoặc tai nạn thì 0 điểm.
- Điểm số và thời gian của từng tay đua sẽ được cộng dồn giữa các chặng để quyết định giải cá nhân và giải đồng đội của mùa giải.

Anh/chị hãy thực hiện modul "**Xem BXH các tay đua**" với các bước sau đây: Ban tổ chức (BTC) chọn chức năng thống kê → Chọn xem bảng xếp hạng các tay đua hiện tại → Hệ thống hiện lên danh sách các tay đua theo dạng bảng, mỗi dòng chứa: Tên tay đua, quốc tịch, tên đội đua, tổng điểm sau các chặng, tổng thời gian sau các chặng. Kết quả sắp xếp theo thứ tự giảm dần của tổng điểm, sau đó là thứ tự tăng dần tổng thời gian. NV click vào 1 dòng của 1 tay đua → hệ thống hiện lên chi tiết kết quả từng chặng đua của tay đua đó, mỗi chặng trên 1 dòng: tên chặng, thứ hạng về đích, số điểm, thời gian về đích.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 40**  
**Thời gian làm bài: 60 phút**

Ban tổ chức đua xe công thức 1 (F1) đặt hàng anh/chị phát triển một phần mềm quản lý kết quả giải đua với mô tả như sau:

- Mỗi năm có một giải. Một giải bao gồm nhiều chặng đua diễn ra trên khắp thế giới (Mã chặng, tên, số vòng đua, địa điểm, thời gian, mô tả).
- Mỗi giải có nhiều đội đua tham gia (Mã, tên, hãng, mô tả).
- Mỗi đội đua có nhiều tay đua (mã, tên, ngày sinh, quốc tịch, tiểu sử). Nhưng ở mỗi chặng đua, mỗi đội chỉ được phép cho tối đa 2 tay đua tham dự.
- Mỗi chặng đua, kết quả xếp theo thứ tự về đích (thời gian) và điểm số chỉ được tính cho top 10 người về đích sớm nhất, lần lượt theo các thứ tự về đích là 25, 18, 15, 12, 10, 8, 6, 4, 2, 1.
- Nếu tay đua nằm trong top 10 nhưng không về đích do bỏ cuộc hoặc tai nạn thì 0 điểm.
- Điểm số và thời gian của từng tay đua sẽ được cộng dồn giữa các chặng để quyết định giải cá nhân và giải đồng đội của mùa giải.

Anh/chị hãy thực hiện modul "**Xem BXH các đội đua**" với các bước sau đây: Ban tổ chức (BTC) chọn chức năng thống kê → Chọn xem bảng xếp hạng các đội đua hiện tại → Hệ thống hiện lên danh sách các đội đua theo dạng bảng, mỗi dòng chứa: Tên đội đua, hãng, tổng điểm các tay đua của đội sau các chặng, tổng thời gian sau các chặng. Kết quả sắp xếp theo thứ tự giảm dần của tổng điểm, sau đó là thứ tự tăng dần tổng thời gian. NV click vào 1 dòng của 1 đội đua → hệ thống hiện lên kết quả chi tiết cho từng chặng của đội đua đó, mỗi chặng trên 1 dòng: tên chặng, tổng số điểm, tổng thời gian của 2 tay đua trong đội.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 41**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu anh/chị phát triển một phần mềm quản lý cho thuê truyện ở một cửa hàng chuyên cho thuê truyện với mô tả như sau:

- Cửa hàng có nhiều đầu truyện khác nhau. Mỗi đầu truyện có số lượng khác nhau và giá thuê khác nhau (giá thuê theo ngày).
- Mỗi đầu truyện có thể được mượn bởi nhiều khách hàng khác nhau. Mỗi khách hàng mỗi lần mượn được mượn nhiều đầu truyện khác nhau.
- Mỗi lần mượn, khách hàng được nhận một phiếu mượn. Trong đó, dòng đầu ghi tên khách hàng và ngày mượn. Thông tin mỗi đầu truyện mượn được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, giá thuê. Dòng cuối cùng ghi số lượng đầu truyện mượn.
- Khi trả truyện, khách hàng được nhận hóa đơn trả. Trong đó, dòng đầu ghi tên khách hàng và ngày thanh toán. Thông tin mỗi đầu truyện trả được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, ngày mượn, ngày trả, giá thuê, thành tiền. Nếu bị phạt thì có thêm cột số tiền phạt. Dòng cuối cùng ghi tổng số tiền thanh toán.

Anh/chị hãy thực hiện modul "**Quản lý đầu truyện**" cho phép quản lý (QL) thực hiện thêm, sửa, xóa thông tin đầu truyện với mô tả chi tiết nghiệp vụ: QL chọn menu quản lý đầu truyện → trang quản lý hiện ra → QL chọn chức năng sửa thông tin đầu truyện → giao diện tìm đầu truyện theo tên hiện ra → QL nhập tên đầu truyện và click tìm kiếm → danh sách các đầu truyện có tên chứa từ khóa hiện ra → QL chọn sửa một đầu truyện → giao diện sửa đầu truyện hiện ra với các thông tin của đầu truyện đã chọn → QL nhập một số thông tin thay đổi và click cập nhật → hệ thống lưu thông tin vào CSDL và thông báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 42**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu anh/chị phát triển một phần mềm quản lí cho thuê truyện ở một cửa hàng chuyên cho thuê truyện với mô tả như sau:

- Cửa hàng có nhiều đầu truyện khác nhau. Mỗi đầu truyện có số lượng khác nhau và giá thuê khác nhau (giá thuê theo ngày).
- Mỗi đầu truyện có thể được mượn bởi nhiều khách hàng khác nhau. Mỗi khách hàng mỗi lần mượn được mượn nhiều đầu truyện khác nhau.
- Mỗi lần mượn, khách hàng được nhận một phiếu mượn. Trong đó, dòng đầu ghi tên khách hàng và ngày mượn. Thông tin mỗi đầu truyện mượn được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, giá thuê. Dòng cuối cùng ghi số lượng đầu truyện mượn.
- Khi trả truyện, khách hàng được nhận hóa đơn trả. Trong đó, dòng đầu ghi tên khách hàng và ngày thanh toán. Thông tin mỗi đầu truyện trả được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, ngày mượn, ngày trả, giá thuê, thành tiền. Nếu bị phạt thì có thêm cột số tiền phạt. Dòng cuối cùng ghi tổng số tiền thanh toán.

Anh/chị hãy thực hiện modul "**Cho thuê truyện**" với các bước sau đây: Sau khi chọn được các truyện để thuê mượn, khách hàng (KH) cầm đến quầy nhân viên (NV) thu ngân làm phiếu mượn. NV nhập tên KH và tìm kiếm → Hệ thống trả về danh sách các KH có tên vừa nhập → NV click chọn tên KH trong danh sách (nếu KH mượn lần đầu thì nhập mới) → Hệ thống hiện giao diện thêm truyện mượn vào phiếu: Với mỗi đầu truyện, NV click chọn tìm truyện theo tên → nhập tên truyện + click tìm → hệ thống hiện lên danh sách các đầu truyện có tên vừa nhập → NV click chọn dòng đúng với quyển truyện do KH chọn thuê → Hệ thống thêm 1 dòng tương ứng với đầu truyện đó vào phiếu thuê mượn như mô tả. Khi hết các đầu truyện do KH chọn thuê, NV click tạo phiếu mượn → Hệ thống lưu vào CSDL và hiển thị phiếu mượn lên màn hình → NV click in ra → Hệ thống in phiếu mượn ra cho KH.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này



**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 43**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu anh/chị phát triển một phần mềm quản lí cho thuê truyện ở một cửa hàng chuyên cho thuê truyện với mô tả như sau:

- Cửa hàng có nhiều đầu truyện khác nhau. Mỗi đầu truyện có số lượng khác nhau và giá thuê khác nhau (giá thuê theo ngày).
- Mỗi đầu truyện có thể được mượn bởi nhiều khách hàng khác nhau. Mỗi khách hàng mỗi lần mượn được mượn nhiều đầu truyện khác nhau.
- Mỗi lần mượn, khách hàng được nhận một phiếu mượn. Trong đó, dòng đầu ghi tên khách hàng và ngày mượn. Thông tin mỗi đầu truyện mượn được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, giá thuê. Dòng cuối cùng ghi số lượng đầu truyện mượn.
- Khi trả truyện, khách hàng được nhận hóa đơn trả. Trong đó, dòng đầu ghi tên khách hàng và ngày thanh toán. Thông tin mỗi đầu truyện trả được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, ngày mượn, ngày trả, giá thuê, thành tiền. Nếu bị phạt thì có thêm cột số tiền phạt. Dòng cuối cùng ghi tổng số tiền thanh toán.

Anh/chị hãy thực hiện modul "**Khách hàng trả truyện và thanh toán**" với các bước sau đây: Khi KH đem truyện đến trả, NV chọn menu tìm danh sách truyện mượn theo tên KH → nhập tên KH+click tìm kiếm → hệ thống hiển thị danh sách các KH có tên vừa nhập → NV chọn tên KH đúng với thông tin KH hiện tại → hệ thống hiện lên danh sách các đầu truyện mà KH đó đang mượn, mỗi đầu truyện trên một dòng với đầy đủ thông tin về đầu truyện, ngày mượn, giá mượn, và số tiền thuê tính đến ngày đang trả, cột cuối cùng là ô tích chọn trả → NV click vào nút chọn trả cho các đầu truyện mà KH đem trả (có thể không trả hết 1 lần), nhập tình trạng sách và tiền phạt nếu có, cuối cùng click nút thanh toán → hệ thống hiện hóa đơn đầy đủ thông tin khách hàng + 1 bảng danh sách các đầu truyện trả như mô tả trên + dòng cuối là tổng số tiền trả → NV click xác nhận → hệ thống cập nhật vào CSDL.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 44**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu anh/chị phát triển một phần mềm quản lí cho thuê truyện ở một cửa hàng chuyên cho thuê truyện với mô tả như sau:

- Cửa hàng có nhiều đầu truyện khác nhau. Mỗi đầu truyện có số lượng khác nhau và giá thuê khác nhau (giá thuê theo ngày).
- Mỗi đầu truyện có thể được mượn bởi nhiều khách hàng khác nhau. Mỗi khách hàng mỗi lần mượn được mượn nhiều đầu truyện khác nhau.
- Mỗi lần mượn, khách hàng được nhận một phiếu mượn. Trong đó, dòng đầu ghi tên khách hàng và ngày mượn. Thông tin mỗi đầu truyện mượn được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, giá thuê. Dòng cuối cùng ghi số lượng đầu truyện mượn.
- Khi trả truyện, khách hàng được nhận hóa đơn trả. Trong đó, dòng đầu ghi tên khách hàng và ngày thanh toán. Thông tin mỗi đầu truyện trả được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, ngày mượn, ngày trả, giá thuê, thành tiền. Nếu bị phạt thì có thêm cột số tiền phạt. Dòng cuối cùng ghi tổng số tiền thanh toán.

Anh/chị hãy thực hiện modul "***Thống kê truyện được mượn nhiều***" với các bước sau đây: QL chọn menu thống kê đầu truyện được mượn nhiều → Nhập khoảng thời gian (ngày bắt đầu – kết thúc) thống kê → Hệ thống hiển thị danh sách các đầu truyện được mượn nhiều theo dạng bảng, mỗi dòng tương ứng với một đầu truyện với đầy đủ thông tin: mã, tên, tác giả, NXB, năm XB, cột tổng số lượt được mượn, cột tổng số tiền thu được. Xếp theo thứ tự giảm dần của cột tổng số lượt mượn, tiếp theo là giảm dần của cột tổng số tiền thu được. NV click vào 1 dòng của 1 truyện → hệ thống hiện lên chi tiết hóa đơn có truyện đó đã mượn, mỗi hóa đơn trên 1 dòng: id, tên khách mượn, ngày giờ mượn, ngày giờ trả, tổng số tiền.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 45**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu anh/chị phát triển một phần mềm quản lí cho thuê truyện ở một cửa hàng chuyên cho thuê truyện với mô tả như sau:

- Cửa hàng có nhiều đầu truyện khác nhau. Mỗi đầu truyện có số lượng khác nhau và giá thuê khác nhau (giá thuê theo ngày).
- Mỗi đầu truyện có thể được mượn bởi nhiều khách hàng khác nhau. Mỗi khách hàng mỗi lần mượn được mượn nhiều đầu truyện khác nhau.
- Mỗi lần mượn, khách hàng được nhận một phiếu mượn. Trong đó, dòng đầu ghi tên khách hàng và ngày mượn. Thông tin mỗi đầu truyện mượn được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, giá thuê. Dòng cuối cùng ghi số lượng đầu truyện mượn.
- Khi trả truyện, khách hàng được nhận hóa đơn trả. Trong đó, dòng đầu ghi tên khách hàng và ngày thanh toán. Thông tin mỗi đầu truyện trả được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, ngày mượn, ngày trả, giá thuê, thành tiền. Nếu bị phạt thì có thêm cột số tiền phạt. Dòng cuối cùng ghi tổng số tiền thanh toán.

Anh/chị hãy thực hiện modul "**Thống kê khách hàng mượn nhiều**" với các bước sau đây: QL chọn menu thống kê khách hàng mượn nhiều → Nhập khoảng thời gian (ngày bắt đầu – kết thúc) thống kê → hệ thống hiển thị danh sách KH mượn nhiều theo dạng bảng, mỗi dòng tương ứng với một KH với đầy đủ thông tin: mã, tên, số CMT, số đt, địa chỉ, tiếp theo là cột tổng số lượt mượn, cột tổng số tiền đã trả. Xếp theo chiều giảm dần của tổng số lượt mượn, tiếp theo là chiều giảm dần của tổng số tiền trả. NV click vào 1 dòng của 1 khách hàng → hệ thống hiện lên chi tiết các hóa đơn khách hàng đã mượn, mỗi hóa đơn trên 1 dòng: ngày mượn, tổng số sách mượn, tổng số tiền thanh toán.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 46**  
**Thời gian làm bài: 60 phút**

Khách hàng yêu cầu anh/chị phát triển một phần mềm quản lí cho thuê truyện ở một cửa hàng chuyên cho thuê truyện với mô tả như sau:

- Cửa hàng có nhiều đầu truyện khác nhau. Mỗi đầu truyện có số lượng khác nhau và giá thuê khác nhau (giá thuê theo ngày).
- Mỗi đầu truyện có thể được mượn bởi nhiều khách hàng khác nhau. Mỗi khách hàng mỗi lần mượn được mượn nhiều đầu truyện khác nhau.
- Mỗi lần mượn, khách hàng được nhận một phiếu mượn. Trong đó, dòng đầu ghi tên khách hàng và ngày mượn. Thông tin mỗi đầu truyện mượn được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, giá thuê. Dòng cuối cùng ghi số lượng đầu truyện mượn.
- Khi trả truyện, khách hàng được nhận hóa đơn trả. Trong đó, dòng đầu ghi tên khách hàng và ngày thanh toán. Thông tin mỗi đầu truyện trả được ghi trên một dòng: tên, tác giả, nhà xuất bản, năm xuất bản, ngày mượn, ngày trả, giá thuê, thành tiền. Nếu bị phạt thì có thêm cột số tiền phạt. Dòng cuối cùng ghi tổng số tiền thanh toán.

Anh/chị hãy thực hiện modul "**Thống kê doanh thu**" với các bước sau đây: QL chọn menu thống kê doanh thu theo thời gian (tháng, quý, năm) → hệ thống hiện ô chọn thống kê theo tháng, quý, hoặc năm → QL click chọn theo tháng → hệ thống hiện lên thống kê doanh thu theo tháng dưới dạng bảng, mỗi dòng tương ứng với 1 tháng (tương ứng là quý, năm): tên tháng, tổng doanh thu. Sắp xếp theo chiều thời gian tháng (tương ứng là quý, năm) gần nhất đến tháng (tương ứng là quý, năm) cũ nhất. NV click vào 1 dòng → hệ thống hiện lên chi tiết các hóa đơn trong khoảng thời gian của dòng đấy, mỗi hóa đơn trên 1 dòng: id, tên khách hàng, ngày mượn, tổng số truyện mượn, tổng số tiền của hóa đơn.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

## Ngân hàng câu hỏi thi môn: Công nghệ phần mềm

### Đề số 47

Thời gian làm bài: 60 phút

Chuỗi nhà hàng đồ ăn nhanh Lotteria đặt hàng anh chị phát triển một phần mềm giúp họ quản lý phân công và chấm công nhân viên làm thêm theo giờ (parttime) tại chuỗi cửa hàng của họ với mô tả như sau:

- Chuỗi nhà hàng có nhiều nhà hàng. Mỗi nhà hàng có nhiều nhân viên làm theo giờ. Mỗi ngày làm việc có 2 ca, ca 1 từ 8-16h, ca 2 từ 16-24h. Mức tiền công theo giờ là giống nhau cho tất cả nhân viên làm theo giờ.
- Mỗi nhân viên, sau khi kí hợp đồng, được đăng kí những buổi nào rảnh để có thể đến làm việc. Số buổi có thể làm việc trong mỗi tuần mà mỗi nhân viên đăng kí phải đạt ngưỡng tối thiểu theo quy định. Thông tin này có thể thay đổi hàng tuần, trước khi lên lịch làm việc cho tuần tiếp theo.
- Quản lý sẽ dựa trên lịch đăng kí của từng nhân viên để lên lịch cho tuần tiếp theo. Đảm bảo mỗi ca có đủ N nhân viên làm việc. Nếu có ca nào đó mà số nhân viên đăng kí lớn hơn N, thì ưu tiên những nhân viên đang có số giờ làm ít hơn xếp trước. Lịch tuần tiếp theo sẽ được thông báo cho toàn bộ nhân viên để tiện chuẩn bị.
- Khi đến làm việc, nhân viên quét thẻ checkin giờ vào làm, khi về, nhân viên quét thẻ checkout để về.
- Tiền lương nhân viên tính theo số giờ thực làm của nhân viên và được trả theo tuần. Ca nào nhân viên làm quá 8h thì mức tiền công cho phần thời gian đội thêm được tính thêm 20%. ca nào nhân viên đến muộn hoặc về sớm thì thời gian vắng mặt sẽ bị trừ tiền đội thêm 50%.

Anh/chị hãy thực hiện modul "**Quản lý thông tin nhân viên**" với các chức năng thêm, sửa, xóa thông tin NV. Chức năng sửa thông tin NV được mô tả như sau: QL chọn chức năng sửa thông tin NV → Giao diện tìm NV theo tên hiện lên → QL nhập tên hoặc một phần tên NV và click tìm → Giao diện hiện tất cả các NV có tên chứa chuỗi từ khóa vừa nhập, mỗi NV trên 1 dòng → QL click chọn dòng đúng NV cần sửa → Giao diện sửa thông tin NV hiện lên với đầy đủ các ô đã có sẵn thông tin hiện thời: tên, địa chỉ, số điện thoại, email → QL chọn sửa lại một số thông tin và click submit → Hệ thống lưu lại và báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 48**  
**Thời gian làm bài: 60 phút**

Chuỗi nhà hàng đồ ăn nhanh Lotteria đặt hàng anh chị phát triển một phần mềm giúp họ quản lý phân công và chấm công nhân viên làm thêm theo giờ (parttime) tại chuỗi cửa hàng của họ với mô tả như sau:

- Chuỗi nhà hàng có nhiều nhà hàng. Mỗi nhà hàng có nhiều nhân viên làm theo giờ. Mỗi ngày làm việc có 2 ca, ca 1 từ 8-16h, ca 2 từ 16-24h. Mức tiền công theo giờ là giống nhau cho tất cả nhân viên làm theo giờ.
- Mỗi nhân viên, sau khi kí hợp đồng, được đăng kí những buổi nào rảnh để có thể đến làm việc. Số buổi có thể làm việc trong mỗi tuần mà mỗi nhân viên đăng kí phải đạt ngưỡng tối thiểu theo quy định. Thông tin này có thể thay đổi hàng tuần, trước khi lên lịch làm việc cho tuần tiếp theo.
- Quản lý sẽ dựa trên lịch đăng kí của từng nhân viên để lên lịch cho tuần tiếp theo. Đảm bảo mỗi ca có đủ N nhân viên làm việc. Nếu có ca nào đó mà số nhân viên đăng kí lớn hơn N, thì ưu tiên những nhân viên đang có số giờ làm ít hơn xếp trước. Lịch tuần tiếp theo sẽ được thông báo cho toàn bộ nhân viên để tiện chuẩn bị.
- Khi đến làm việc, nhân viên quét thẻ checkin giờ vào làm, khi về, nhân viên quét thẻ checkout để về.
- Tiền lương nhân viên tính theo số giờ thực làm của nhân viên và được trả theo tuần. Ca nào nhân viên làm quá 8h thì mức tiền công cho phần thời gian đội thêm được tính thêm 20%. ca nào nhân viên đến muộn hoặc về sớm thì thời gian vắng mặt sẽ bị trừ tiền đội thêm 50%.

Anh/chị hãy thực hiện modul "**Đăng kí ca làm tuần tới**" với các bước sau đây: QL chọn chức năng đăng kí ca làm tuần tới cho NV → Giao diện tìm NV hiện lên → QL nhập tên NV hoặc một phần tên NV và click tìm → Giao diện hiện lên danh sách các NV có tên chứa từ khóa vừa nhập → Giao diện đăng kí ca làm tuần tới cho NV hiện lên, chứa thông tin NV và 1 bảng có 7 dòng tương ứng 7 ngày của tuần tới, mỗi dòng có 2 ô chọn tương ứng với ca → QL click vào các ô tương ứng với các ca mà NV đăng kí làm và click lưu → Hệ thống lưu lại và báo thành công.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

## Ngân hàng câu hỏi thi môn: Công nghệ phần mềm

### Đề số 49

#### Thời gian làm bài: 60 phút

Chuỗi nhà hàng đồ ăn nhanh Lotteria đặt hàng anh chị phát triển một phần mềm giúp họ quản lý phân công và chấm công nhân viên làm thêm theo giờ (parttime) với mô tả như sau:

- Chuỗi nhà hàng có nhiều nhà hàng. Mỗi nhà hàng có nhiều nhân viên làm theo giờ. Mỗi ngày làm việc có 2 ca, ca 1 từ 8-16h, ca 2 từ 16-24h. Mức tiền công theo giờ là giống nhau cho tất cả nhân viên làm theo giờ.
- Mỗi nhân viên, sau khi kí hợp đồng, được đăng kí những buổi nào rảnh để có thể đến làm việc. Số buổi có thể làm việc trong mỗi tuần mà mỗi nhân viên đăng kí phải đạt ngưỡng tối thiểu theo quy định. Thông tin này có thể thay đổi hàng tuần, trước khi lên lịch làm việc cho tuần tiếp theo.
- Quản lý sẽ dựa trên lịch đăng kí của từng nhân viên để lên lịch cho tuần tiếp theo. Đảm bảo mỗi ca có đủ N nhân viên làm việc. Nếu có ca nào đó mà số nhân viên đăng kí lớn hơn N, thì ưu tiên những nhân viên đang có số giờ làm ít hơn xếp trước. Lịch tuần tiếp theo sẽ được thông báo cho toàn bộ nhân viên để tiện chuẩn bị.
- Khi đến làm việc, nhân viên quét thẻ checkin giờ vào làm, khi về, nhân viên quét thẻ checkout để về.
- Tiền lương nhân viên tính theo số giờ thực làm của nhân viên và được trả theo tuần. Ca nào nhân viên làm quá 8h thì mức tiền công cho phần thời gian đội thêm được tính thêm 20%. ca nào nhân viên đến muộn hoặc về sớm thì thời gian vắng mặt sẽ bị trừ tiền đội thêm 50%.

Anh/chị hãy thực hiện modul "**Lên lịch làm việc tuần tới**" với các bước sau đây: QL chọn chức năng lên lịch làm việc tuần tới cho nhân viên 1 nhà hàng → Giao diện lên lịch hiện lên gồm một bảng có 7 dòng tương ứng 7 ngày của tuần tới, mỗi dòng có 2 cột tương ứng 2 ca của ngày. Mỗi cột chứa tên các NV đã chọn cho ca đó → QL click chọn vào 1 ca → Giao diện hiện lên danh sách các NV đã đăng kí làm việc cho ca đó và chưa được xếp làm cho ca đó, mỗi NV trên 1 dòng: tên, số điện thoại, tổng giờ đã lên lịch cho tuần tới, sắp xếp theo chiều tăng dần của tổng giờ đã lên lịch cho tuần tới → QL click chọn một số NV và click nút chọn → Giao diện quay về trang lên lịch với thông tin các NV được chọn được thêm vào cột của ca tương ứng → QL lặp lại các bước chọn trên cho đến hết số ca của tuần tới và click lưu → Hệ thống lưu lại và thông báo thành công, đồng thời in lịch ra để QL phát cho từng NV.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

**Ngân hàng câu hỏi thi môn: Công nghệ phần mềm**  
**Đề số 50**  
**Thời gian làm bài: 60 phút**

Chuỗi nhà hàng đồ ăn nhanh Lotteria đặt hàng anh chị phát triển một phần mềm giúp họ quản lý phân công và chấm công nhân viên làm thêm theo giờ (parttime) tại chuỗi cửa hàng của họ với mô tả như sau:

- Chuỗi nhà hàng có nhiều nhà hàng. Mỗi nhà hàng có nhiều nhân viên làm theo giờ. Mỗi ngày làm việc có 2 ca, ca 1 từ 8-16h, ca 2 từ 16-24h. Mức tiền công theo giờ là giống nhau cho tất cả nhân viên làm theo giờ.
- Mỗi nhân viên, sau khi kí hợp đồng, được đăng kí những buổi nào rảnh để có thể đến làm việc. Số buổi có thể làm việc trong mỗi tuần mà mỗi nhân viên đăng kí phải đạt ngưỡng tối thiểu theo quy định. Thông tin này có thể thay đổi hàng tuần, trước khi lên lịch làm việc cho tuần tiếp theo.
- Quản lý sẽ dựa trên lịch đăng kí của từng nhân viên để lên lịch cho tuần tiếp theo. Đảm bảo mỗi ca có đủ N nhân viên làm việc. Nếu có ca nào đó mà số nhân viên đăng kí lớn hơn N, thì ưu tiên những nhân viên đang có số giờ làm ít hơn xếp trước. Lịch tuần tiếp theo sẽ được thông báo cho toàn bộ nhân viên để tiện chuẩn bị.
- Khi đến làm việc, nhân viên quét thẻ checkin giờ vào làm, khi về, nhân viên quét thẻ checkout để về.
- Tiền lương nhân viên tính theo số giờ thực làm của nhân viên và được trả theo tuần. Ca nào nhân viên làm quá 8h thì mức tiền công cho phần thời gian đội thêm được tính thêm 20%. ca nào nhân viên đến muộn hoặc về sớm thì thời gian vắng mặt sẽ bị trừ tiền đội thêm 50%.

Anh/chị hãy thực hiện modul "**Checkin/Checkout**" với các bước sau đây: Checkin và checkout có thể do NV quét thẻ, hoặc do QL trực tiếp cập nhật trên máy tính: QL chọn chức năng checkin (hoặc checkout) → Giao diện nhập mã NV hiện lên → QL nhập mã NV và click submit → Hệ thống lưu và báo thời điểm checkin (checkout) của NV là thời điểm hiện tại.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này



## Ngân hàng câu hỏi thi môn: Công nghệ phần mềm

### Đề số 51

#### Thời gian làm bài: 60 phút

Chuỗi nhà hàng đồ ăn nhanh Lotteria đặt hàng anh chị phát triển một phần mềm giúp họ quản lý phân công và chấm công nhân viên làm thêm theo giờ (parttime) với mô tả như sau:

- Chuỗi nhà hàng có nhiều nhà hàng. Mỗi nhà hàng có nhiều nhân viên làm theo giờ. Mỗi ngày làm việc có 2 ca, ca 1 từ 8-16h, ca 2 từ 16-24h. Mức tiền công theo giờ là giống nhau cho tất cả nhân viên làm theo giờ.
- Mỗi nhân viên, sau khi kí hợp đồng, được đăng kí những buổi nào rảnh để có thể đến làm việc. Số buổi có thể làm việc trong mỗi tuần mà mỗi nhân viên đăng kí phải đạt ngưỡng tối thiểu theo quy định. Thông tin này có thể thay đổi hàng tuần, trước khi lên lịch làm việc cho tuần tiếp theo.
- Quản lý sẽ dựa trên lịch đăng kí của từng nhân viên để lên lịch cho tuần tiếp theo. Đảm bảo mỗi ca có đủ N nhân viên làm việc. Nếu có ca nào đó mà số nhân viên đăng kí lớn hơn N, thì ưu tiên những nhân viên đang có số giờ làm ít hơn xếp trước. Lịch tuần tiếp theo sẽ được thông báo cho toàn bộ nhân viên để tiện chuẩn bị.
- Khi đến làm việc, nhân viên quét thẻ checkin giờ vào làm, khi về, nhân viên quét thẻ checkout để về.
- Tiền lương nhân viên tính theo số giờ thực làm của nhân viên và được trả theo tuần. Ca nào nhân viên làm quá 8h thì mức tiền công cho phần thời gian đội thêm được tính thêm 20%. ca nào nhân viên đến muộn hoặc về sớm thì thời gian vắng mặt sẽ bị trừ tiền đội thêm 50%.

Anh/chị hãy thực hiện modul "**Tính công tuần này**" với các bước sau đây: QL chọn chức năng tính tiền công cho NV trong tuần → Giao diện tính công hiện lên với ô nhập khoảng thời gian tính công → QL nhập ngày bắt đầu, ngày kết thúc của tuần vừa rồi → Giao diện hiện lên danh sách bảng tiền công cho tất cả các NV trong tuần đó, mỗi NV trên 1 dòng, xếp theo thứ tự tên: mã, tên, số đt, tổng số giờ làm trong ca, tổng tiền trong ca, tổng số giờ thừa ca, tổng tiền thừa ca, tổng số giờ đi chậm về sớm, tổng số tiền bị phạt, tổng tiền thực nhận cuối cùng → QL click chọn vào 1 dòng để xem chi tiết → Giao diện hiện lên bảng thống kê chi tiết giờ làm của NV được chọn trong tuần đó, mỗi dòng tương ứng 1 ca làm việc, xếp theo thứ tự thời gian: thứ, ngày, ca, giờ checkin, giờ checkout, số giờ trong ca, số tiền trong ca, số giờ thừa ca, số tiền thừa ca, số giờ đi chậm về sớm, số tiền bị phạt, tổng tiền thực nhận của ca.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

## Ngân hàng câu hỏi thi môn: Công nghệ phần mềm

### Đề số 52

#### Thời gian làm bài: 60 phút

Chuỗi nhà hàng đồ ăn nhanh Lotteria đặt hàng anh chị phát triển một phần mềm giúp họ quản lý phân công và chấm công nhân viên làm thêm theo giờ (parttime) với mô tả như sau:

- Chuỗi nhà hàng có nhiều nhà hàng. Mỗi nhà hàng có nhiều nhân viên làm theo giờ. Mỗi ngày làm việc có 2 ca, ca 1 từ 8-16h, ca 2 từ 16-24h. Mức tiền công theo giờ là giống nhau cho tất cả nhân viên làm theo giờ.
- Mỗi nhân viên, sau khi kí hợp đồng, được đăng kí những buổi nào rảnh để có thể đến làm việc. Số buổi có thể làm việc trong mỗi tuần mà mỗi nhân viên đăng kí phải đạt ngưỡng tối thiểu theo quy định. Thông tin này có thể thay đổi hàng tuần, trước khi lên lịch làm việc cho tuần tiếp theo.
- Quản lý sẽ dựa trên lịch đăng kí của từng nhân viên để lên lịch cho tuần tiếp theo. Đảm bảo mỗi ca có đủ N nhân viên làm việc. Nếu có ca nào đó mà số nhân viên đăng kí lớn hơn N, thì ưu tiên những nhân viên đang có số giờ làm ít hơn xếp trước. Lịch tuần tiếp theo sẽ được thông báo cho toàn bộ nhân viên để tiện chuẩn bị.
- Khi đến làm việc, nhân viên quét thẻ checkin giờ vào làm, khi về, nhân viên quét thẻ checkout để về.
- Tiền lương nhân viên tính theo số giờ thực làm của nhân viên và được trả theo tuần. Ca nào nhân viên làm quá 8h thì mức tiền công cho phần thời gian đội thêm được tính thêm 20%. ca nào nhân viên đến muộn hoặc về sớm thì thời gian vắng mặt sẽ bị trừ tiền đội thêm 50%.

Anh/chị hãy thực hiện modul "**Thống kê nhân viên làm nhiều**" với các bước sau đây: QL chọn chức năng thống kê NV làm nhiều → Giao diện thống kê hiện lên với ô nhập khoảng thời gian thống kê → QL nhập ngày bắt đầu, ngày kết thúc của thời gian thống kê → Giao diện hiện lên danh sách bảng thống kê cho tất cả các NV trong khoảng thời gian đó, mỗi NV trên 1 dòng, xếp theo thứ tự tổng số giờ làm giảm dần: mã, tên, số đt, tổng số giờ làm trong ca, tổng số giờ thừa ca, tổng số giờ đi chậm về sớm, tổng số giờ thực làm cuối cùng, tổng số tiền thực nhận cuối cùng → QL click chọn vào 1 dòng để xem chi tiết → Giao diện hiện lên bảng thống kê chi tiết giờ làm của NV được chọn trong khoảng thời gian đó, mỗi dòng tương ứng 1 ca làm việc, xếp theo thứ tự thời gian: thứ, ngày, ca, giờ checkin, giờ checkout, số giờ trong ca, số giờ thừa ca, số giờ đi chậm về sớm, tổng thời gian thực làm, tổng tiền thực nhận của ca.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này

## Ngân hàng câu hỏi thi môn: Công nghệ phần mềm

### Đề số 53

#### Thời gian làm bài: 60 phút

Chuỗi nhà hàng đồ ăn nhanh Lotteria đặt hàng anh chị phát triển một phần mềm giúp họ quản lý phân công và chấm công nhân viên làm thêm theo giờ (parttime) với mô tả như sau:

- Chuỗi nhà hàng có nhiều nhà hàng. Mỗi nhà hàng có nhiều nhân viên làm theo giờ. Mỗi ngày làm việc có 2 ca, ca 1 từ 8-16h, ca 2 từ 16-24h. Mức tiền công theo giờ là giống nhau cho tất cả nhân viên làm theo giờ.
- Mỗi nhân viên, sau khi kí hợp đồng, được đăng kí những buổi nào rảnh để có thể đến làm việc. Số buổi có thể làm việc trong mỗi tuần mà mỗi nhân viên đăng kí phải đạt ngưỡng tối thiểu theo quy định. Thông tin này có thể thay đổi hàng tuần, trước khi lên lịch làm việc cho tuần tiếp theo.
- Quản lý sẽ dựa trên lịch đăng kí của từng nhân viên để lên lịch cho tuần tiếp theo. Đảm bảo mỗi ca có đủ N nhân viên làm việc. Nếu có ca nào đó mà số nhân viên đăng kí lớn hơn N, thì ưu tiên những nhân viên đang có số giờ làm ít hơn xếp trước. Lịch tuần tiếp theo sẽ được thông báo cho toàn bộ nhân viên để tiện chuẩn bị.
- Khi đến làm việc, nhân viên quét thẻ checkin giờ vào làm, khi về, nhân viên quét thẻ checkout để về.
- Tiền lương nhân viên tính theo số giờ thực làm của nhân viên và được trả theo tuần. Ca nào nhân viên làm quá 8h thì mức tiền công cho phần thời gian đội thêm được tính thêm 20%. ca nào nhân viên đến muộn hoặc về sớm thì thời gian vắng mặt sẽ bị trừ tiền đội thêm 50%.

Anh/chị hãy thực hiện modul "**Thống kê nhân viên đúng giờ**" với các bước sau đây: QL chọn chức năng thống kê NV đúng giờ → Giao diện thống kê hiện lên với ô nhập khoảng thời gian thống kê → QL nhập ngày bắt đầu, ngày kết thúc của thời gian thống kê → Giao diện hiện lên danh sách bảng thống kê cho tất cả các NV trong khoảng thời gian đó, mỗi NV trên 1 dòng, xếp theo thứ tự tăng dần của tổng số giờ đi muộn về sớm: mã, tên, số đt, tổng số giờ thực làm, tổng tiền thực nhận, tổng số giờ đi chậm về sớm, tổng số tiền phạt → QL click chọn vào 1 dòng để xem chi tiết → Giao diện hiện lên bảng thống kê chi tiết giờ làm của NV được chọn trong khoảng thời gian đó, mỗi dòng tương ứng 1 ca làm việc, xếp theo thứ tự thời gian: thứ, ngày, ca, giờ checkin, giờ checkout, số giờ thực làm, số tiền thực nhận, số giờ đi chậm về sớm, số tiền phạt.

1. Viết một scenario chuẩn cho use case này
2. Trích và vẽ biểu đồ các lớp thực thể của toàn bộ hệ thống
3. Thiết kế tĩnh: thiết kế giao diện và vẽ biểu đồ lớp MVC chi tiết cho modul
4. Thiết kế động: vẽ biểu đồ tuần tự mô tả tuần tự hoạt động của modul
5. Viết một test case chuẩn cho modul này