

Quy định nộp bài tập tuần và BTL

Môn: Thiết kế và xây dựng phần mềm
Giảng viên: TS. Nguyễn Thị Thu Trang

1 Yêu cầu:

1.1 Đặt tên Team và Repository

TKXDPM.20191-YY, trong đó YY là mã nhóm, ví dụ 02, 14.

1.2 Cách thức và thời gian nộp

- Cách thức nộp:
 - Nộp qua bitbucket với tên team và repository được quy định trong phần 1.1
 - Add tài khoản trangntt-student để giáo viên theo dõi và chấm điểm
- Cách đặt tên
 - Các bài tập tuần cần được đặt trong các thư mục tương ứng, bao gồm:
 - RequirementAnalysis
 - ArchitecturalDesign
 - DetailedDesign
 - Programming
 - UnitTest
 - Bài tập lớn được đặt trong thư mục Project
- Thời gian nộp:
 - 22h thứ 7 của tuần có bài tập về nhà

1.3 Nhiệm vụ

- Mỗi thành viên phụ trách 1-2 use case cho các Homework với tiêu chí chia đều công việc cho mỗi thành viên. Mỗi thành viên đều phải tự làm và đẩy lên branch của mình trên bitbucket. Sau đó các thành viên trao đổi với nhau, góp ý, gộp lại để nộp bản cuối cùng của cả nhóm (bao gồm kết quả của các cá nhân) trên nhánh master.
- Các công việc cho các yêu cầu bao gồm tất cả các giai đoạn từ Phân tích yêu cầu, thiết kế (thiết kế kiến trúc, thiết kế chi tiết) đến xây dựng (lập trình, kiểm thử đơn vị),

1.4 Cấu trúc thư mục

- Luôn luôn có file Readme.md trong đó chứa:
 - Phân công công việc cho các thành viên trong nhóm: Ai làm gì
 - Phân công review cho các thành viên trong nhóm: Ai review cho ai, review cái gì (nếu chia nhỏ ra). Tránh review cho nhau (A review cho B, B lại review cho A). Khuyến khích review như sau: A review cho B, B review cho C, C review cho D và D review cho A.
- Kết quả làm:
 - Đối với bài tập dạng mã nguồn:
 - Không cần tạo ra riêng thư mục cho từng người, mà làm chung trong 1 Java project
 - Các thao tác của các thành viên đều được log lại trên hệ thống nên giáo viên biết bạn nào làm phần nào hoặc chỉnh sửa cái gì, lúc nào.
 - Đối với bài tập kiểm thử đơn vị, cần nộp:
 - Các trường hợp kiểm thử và kỹ thuật kiểm thử đã sử dụng

- Project chứa mã nguồn của chương trình và mã nguồn kiểm thử
- Đối với bài tập không phải dạng mã nguồn, kiểm thử đơn vị:
 - Kết quả nộp
 - Nộp bản word theo mẫu (nếu có) và chứa ảnh các biểu đồ (nếu có)
 - Nộp bản vẽ trên asta
 - Mỗi người tự tạo thư mục của riêng mình trong thư mục của tuần đó
=> Chiếm 60%. Trong thư mục đó chứa:
 - Các kết quả làm việc cá nhân của thành viên (40%) tương ứng với phân công công việc trong file Readme.md
 - Các kết quả review của người được phân công review (20%):
Tại chỗ thì dùng Track changes/Comment hoặc phía dưới cùng.
Phần review có kèm theo một file review như trong template.
 - Phía ngoài các thư mục cá nhân (trong thư mục tuần) là kết quả cuối cùng của nhóm, sau khi review và ghép lại (nên để dạng word và chứa tất cả các kết quả cần hoàn thiện trong Homework này) => Chiếm 40%

2 Chi tiết các bài tập tuần / BTL

2.1 Homework01: RequirementAnalysis (Phân tích yêu cầu phần mềm)

- Tạo nhóm, repo như quy định
- Tạo thư mục RequirementAnalysis trong repo
- Tạo thư mục AVM: Vẽ biểu đồ use case và luồng sự kiện cho các use case trong hệ thống máy bán hàng tự động Automated Vending Machine (AVM), deadline 22h thứ 7 ngày 21/9/2019. Biểu đồ use case làm chung và do 1 thành viên trong nhóm push lên. Luồng sự kiện phân chia đều cho các thành viên.
- Tạo thư mục AFC: Thực hiện các bước phân tích yêu cầu cho hệ thống soát vé tự động Automated Fare Controller (AFC), deadline 22h thứ 7 ngày 28/9/2019, đầu ra gồm:
 - Chung: Biểu đồ Use case
 - Riêng:
 - Mỗi thành viên đặc tả ít nhất 2 use case
 - Từ điển thuật ngữ: Phân chia công việc để làm
 - Đặc tả phụ trợ: Phân chia công việc để làm
- Ví dụ về cách tổ chức trong bài tập này, trong mỗi thư mục của mỗi hệ thống, gồm các phần tử sau:
 - Các thư mục tương ứng với tên từng cá nhân
 - RequirementAnalysis.doc/docx là tổng hợp của cả nhóm cho Homework02 trong đó chứa
 - 1. Biểu đồ Use case
 - 2. Đặc tả use case của tất cả các use case mà các thành viên trong nhóm phụ trách
 - 3. Từ điển thuật ngữ
 - 4. Đặc tả phụ trợ
 - Readme.md

2.2 Homework02: Thiết kế kiến trúc

- Chỉnh sửa và cập nhật kết quả Homework01 theo những trao đổi trong buổi học trước nếu cần, chú ý cập nhật luôn vào thư mục RequirementAnalysis và tuân theo mẫu SRS được cung cấp trong thư mục Template.

- Tạo thư mục ArchitectureDesign trong repo. Thực hiện các công việc sau cho hệ thống soát vé tự động AFC:
 - Cá nhân: Mỗi cá nhân thiết kế biểu đồ tương tác cho mỗi use case mình phụ trách (có thể phải phân công thêm nếu Homework01 vẫn chưa phân công hết hoặc có bạn không làm): Biểu đồ tương tác (biểu đồ trình tự hoặc/và biểu đồ giao tiếp), Biểu đồ lớp tổng quan cho từng use case.
 - Nhóm: Gộp các biểu đồ lớp tổng quan cho từng use case thành biểu đồ lớp chung của toàn hệ thống.

2.3 Homework03: Thiết kế chi tiết

- Chỉnh sửa và cập nhật kết quả Homework03 theo những trao đổi trong buổi học trước nếu cần, chú ý cập nhật luôn vào thư mục ArchitectureDesign.
- Tạo thư mục DetailDesign trong repo: Thực hiện các công việc sau cho hệ thống soát vé tự động AFC, đặt trong các thư mục tương ứng. Chú ý các phần thiết kế chi tiết cho từng thành phần (màn hình/giao diện, lớp, thực thể/bảng) cần phân chia cho các thành viên trong nhóm bám sát theo phân công use case từ những bước trước.
 - Thư mục UI: Chứa thiết kế giao diện
 - Thư mục ClassDesign: Chứa biểu đồ lớp thiết kế và thiết kế lớp chi tiết
 - Thư mục DataModeling: Thiết kế dữ liệu: Biểu đồ thực thể liên kết, chuẩn hoá, thiết kế CSDL.