



CTT532 – Phát triển Phần mềm Nguồn mở

Đồ án Cuối kỳ

I. Thông tin chung

Mã số bài tập:	DACK
Thời lượng dự kiến:	50 tiết
Deadline nộp bài:	Tuần 15
Hình thức:	Bài tập nhóm
Hình thức nộp bài:	Nộp qua Moodle môn học
GV phụ trách:	Theo bảng “Phân nhóm Đồ án cuối kỳ”
Thông tin liên lạc với GV:	{lxdinh, tploc, vhphec, htthanh}@fit.hcmus.edu.vn

II. Chuẩn đầu ra cần đạt

L07, L08, L09, L10, L11, L12

III. Mô tả bài tập

Tìm hiểu, cài đặt và phát triển plugin/module cho Eclipse/CMS/forum theo hình thức làm việc nhóm chia sẻ công việc và trao đổi qua GitHub. Đồng thời, tìm hiểu tổng quan về FOSS project tương ứng (được sử dụng trong đồ án).

Mỗi nhóm SV chọn một trong 3 nhóm đề tài sau:

- Phát triển một plugin cho Eclipse để hỗ trợ việc phát triển phần mềm trên Eclipse.
- Xây dựng trang web dựa trên một hệ thống CMS nguồn mở có sẵn.
- Xây dựng một website trên một hệ thống forum nguồn mở có sẵn.

IV. Các yêu cầu & quy định chi tiết cho bài nộp

1. Yêu cầu tìm hiểu project (chung cho các loại đề tài)

Khảo sát các khía cạnh sau trên project nguồn mở mà nhóm chọn để phát triển phần mềm trong đồ án của mình.

- Pháp lý: Các giấy phép / hợp đồng mà các bên cần cung cấp / đồng ý. Khảo sát cả 3 bên: Users, Contributors, Project. Trong đó, nếu có nhiều loại Users và nhiều loại Contributors thì những giấy phép / hợp đồng tương ứng cho từng loại là gì.
- Kinh tế:
 - Users được kinh doanh phần mềm lấy từ project này như thế nào, theo những hình thức nào?
 - Contributors được lợi ích kinh tế gì khi tham gia project?

- Project thu nguồn lợi / hỗ trợ kinh tế từ đâu?
- Xã hội:
 - Tổ chức cộng đồng như thế nào, gồm những loại thành viên nào? Mỗi loại thành viên có quyền hạn & trách nhiệm gì, tham gia những hoạt động gì?
 - Những định hướng vĩ mô, kế hoạch, quyết định được thực hiện như thế nào?
 - Cộng đồng gồm những thành phần xã hội nào (tình nguyện viên tự do, nhân viên công ty nào, sinh viên, nghiên cứu viên, v.v.). Tỷ lệ nhân lực và tỷ lệ đóng góp của từng thành phần (nếu không có số liệu thống kê thì SV tự ước lượng).
- Kỹ thuật:
 - Cơ sở hạ tầng được tổ chức như thế nào: tài liệu, diễn đàn, chat, quản lý lỗi, quản lý mã nguồn.
 - Hệ thống phần mềm được thiết kế như thế nào: bởi ai, theo quy trình gì, theo mô hình gì?
 - Hệ thống module trong mã nguồn: phân chia module thế nào, tương tác ra sao?

2a. Yêu cầu phát triển plugin cho Eclipse

SV tự đề xuất chức năng cụ thể và yêu cầu giáo viên phụ trách duyệt lại.

- Thành phần plugin vào Eclipse Platform:
 - Có **ít nhất 1 view** để thể hiện thông tin liên quan đến plugin (output)
 - Phải có tương tác với người dùng (input qua mouse/keyboard).
 - Các thành phần khác như menu (nên có), editor, dialog box,... là tùy chọn.
- Chỉ push lên GitHub nội dung thư mục plugin, **không** push cả mã nguồn Eclipse. Đảm bảo người sử dụng có thể cài plugin mà không cần sửa chữa mã nguồn Eclipse.

Một số gợi ý cho nhóm đề tài plugin Eclipse:

- Testing plugin: ví dụ xây dựng plugin hỗ trợ cho việc test chương trình Java/C/C++
 - Xây dựng màn hình **động** cho phép tùy chọn nhập vào input với số lượng input khác nhau, expected output (output chuẩn).
 - Load và hiển thị mã nguồn với định dạng giống như trong IDE lập trình. Biên dịch & chạy chương trình để lấy ra output thực.
 - <http://checkstyle.sourceforge.net>
 - Sau đó so sánh hai output và đưa ra kết luận.
 - <http://marketplace.eclipse.org/taxonomy/term/14,31>
- Tooltips:
 - Xây dựng giao diện cho phép tái định nghĩa phần mô tả (tooltip) của từng lệnh/control.
 - Ví dụ: khi đưa con trỏ vào lệnh if, thay vì hiển thị tiếng Anh, thì sẽ được định nghĩa lại để thể hiện bằng tiếng Việt. Với trường hợp chưa được chuyển sang tiếng Việt thì vẫn thể hiện bằng tiếng Anh như mặc định.
- Code Assistant:
 - Xây dựng giao diện để hỗ trợ tạo thêm tùy chỉnh code assistant cho ngôn ngữ Java/C/C++. Trong đó:
 - Hỗ trợ nhắc lệnh (ví dụ: gõ chữ if sẽ có gợi ý toàn bộ cấu trúc if () then {} else {}).

- Cho phép xây dựng các template theo yêu cầu người dung.
- Ví dụ: người dung định nghĩa template tương ứng với từ khóa doc_file. Khi người dung sử dụng từ khoá doc_file thì tự động phát ra template tương ứng đã được định nghĩa trước bao gồm comment đầu file có cả license theo tùy chọn.
- Ví dụ:
 - <http://eclipse.org/recommenders/>
 - <http://marketplace.eclipse.org/category/free-tagging/code-completion>

2b. Yêu cầu xây dựng trang web dựa trên CMS

SV tự đề xuất chức năng cụ thể của hệ thống, hệ thống dự định sử dụng và yêu cầu giáo viên phụ trách duyệt lại.

- Yêu cầu về website:
 - Sử dụng CMS để cài đặt, sử dụng theme sẵn có, hiệu chỉnh, triển khai. Không chấp nhận nhóm sinh viên tự cài đặt hoàn toàn website bán hàng.
 - Phải có ít nhất 1 module/plugin là tìm kiếm và sử dụng của đơn vị thứ 3 (nghĩa là plugin/module đó ko nằm sẵn trong bộ plugin/module do CMS cung cấp)
 - Xây dựng module/plugin đăng tin khuyến mãi (gồm cả 2 phần back-end & front-end). Có 2 loại khuyến mãi:
 - Loại 1: click vào hình ảnh khuyến mãi ở trang chủ, sẽ dẫn đến 1 trang mô tả thông tin chi tiết của khuyến mãi đó
 - Loại 2: click vào hình ảnh khuyến mãi, sẽ dẫn đến 1 trang hiện danh sách các sản phẩm đang giảm giá. Từ trang này, người dùng có thể xem chi tiết sản phẩm hoặc đưa sản phẩm vào giỏ hàng.

2c. Yêu cầu xây dựng website dựa trên forum nguồn mở có sẵn

Tìm hiểu, cài đặt và triển khai một website bán hàng đơn giản, sử dụng các open source diễn đàn (forum) thông dụng như:

- phpBB ([link](#)), MyBB ([link](#)), Simple Machines Forum ([link](#)),
- jforum ([link](#)), jGossip, jsforum ([link](#))
- Community Engine ([link](#))
- SV tự đề xuất chức năng cụ thể của hệ thống, hệ thống dự định sử dụng và yêu cầu giáo viên phụ trách duyệt lại.
- Yêu cầu về website:
 - Triển khai trên máy chủ miễn phí.
 - Cài đặt / hiệu chỉnh và sử dụng giao diện.
 - Xây dựng / tìm kiếm module/plugin để quản lý các banner quảng cáo (trên trang chủ, trên từng chủ đề riêng).
 - Xây dựng module/plugin để lọc spam (gồm cả 2 phần back-end/front-end).
 - Back-end: admin qui định thể nào là spam, xem danh sách các bài spam, xử lý người dùng và nội dung, lịch sử.
 - Front-end: báo cáo spam.

V. Các cột mốc

- Deadline 1: Đặc tả đề tài và kế hoạch thực hiện.
- Deadline 2: Triển khai trang web / sử dụng giao diện. (Cho các đề tài web.)
- Deadline 3: Đặc tả chi tiết về module cần phát triển và thiết kế màn hình (giao diện).
- Deadline 4: Vấn đáp cuối kỳ về phần mềm và bản báo cáo (phần tìm hiểu project & đặc tả đồ án).

VI. Cách đánh giá

Đánh giá tìm hiểu project thông qua:

- Độ đầy đủ của bản khảo sát
- Khả năng nắm vững những gì đã khảo sát
- Độ chính xác của bản khảo sát

Đánh giá phát triển plugin thông qua:

- Chức năng của hệ thống
- Sự hoàn thiện & nắm rõ cách hoạt động của plugin
- Tổ chức, trình bày mã nguồn theo hướng dẫn và khuyến cáo của cộng
- Quá trình và kỹ năng phát triển trên GitHub (commit mã nguồn, issue và wiki)
Lưu ý: **Quá trình phát triển** (history) là quan trọng, chứ không chỉ sản phẩm cuối!
- Giao diện của hệ thống

VII. Tài liệu tham khảo

1. Dành cho Eclipse:

- Plug-in Development Environment (PDE)
project: <http://projects.eclipse.org/projects/eclipse.pde>
- Eclipse Java development tools (JDT) project: <http://www.eclipse.org/jdt/>
- Các plugin có sẵn trên thị trường (tham khảo): <http://marketplace.eclipse.org/>
- Toàn tập về cách viết plugin cho Eclipse nói chung và cho Eclipse Platform nói riêng: http://help.eclipse.org/kepler/nav/2_0
- Các điểm mở rộng (extension point) của JDT:
 - Using the code formatter
http://help.eclipse.org/kepler/topic/org.eclipse.jdt.doc.isv/guide/jdt_api_codeformatter.htm?cp=3_0_0_8
 - Performing code assist on Java code
http://help.eclipse.org/kepler/topic/org.eclipse.jdt.doc.isv/guide/jdt_api_codeassist.htm?cp=3_0_0_6



- Customizing Java
editors: http://help.eclipse.org/kepler/topic/org.eclipse.jdt.doc.isv/guide/jdt_api_editors.htm?cp=3_0_1_3
- Contributing a quick fix and a quick assist for Java
code: http://help.eclipse.org/kepler/topic/org.eclipse.jdt.doc.isv/guide/jdt_api_contributing_a_quickfix.htm?cp=3_0_1_6
- Contributing a clean up and a save action using the clean up extension point: http://help.eclipse.org/kepler/topic/org.eclipse.jdt.doc.isv/guide/jdt_api_contributing_a_cleanup.htm?cp=3_0_1_5
- Tắt
cả: http://help.eclipse.org/kepler/topic/org.eclipse.jdt.doc.isv/reference/extension-points/index.html?cp=3_1_1
- Các bài hướng dẫn viết plugin:
 - <http://users.drew.edu/bburd/Eclipse/CreatingAnEclipsePlugin.html>
 - http://agile.csc.ncsu.edu/SEMaterials/tutorials/plugin_dev/index.html#section1_0
 - <http://www.integratingstuff.com/2010/11/23/writing-an-eclipse-plugin/>

2. Dành cho Web:

- Mã nguồn WordPress (git): <https://GitHub.com/WordPress/WordPress>
- Cách xây dựng plugin
WordPress: https://codex.WordPress.org/Writing_a_Plugin
- Webserver XAMPP: <https://www.apachefriends.org/index.html>
- Webserver WAMP: <http://www.wampserver.com/en/>
- Công cụ git: <http://git-scm.com/>
- Tổ chức mã nguồn
plugin: https://codex.WordPress.org/Determining_Plugin_and_Content_Directories
- Thư viện plugin có sẵn (tham khảo): <https://WordPress.org/plugins/>
- Trang phát triển của WordPress core: <http://make.WordPress.org/core/>
- Khác: http://codex.WordPress.org/Getting_Started_with_WordPress

VIII. Các quy định khác

- Sử dụng GitHub để chia sẻ mã nguồn phát triển; Đánh dấu các phiên bản bằng tag, và các **phiên bản nộp bài** phải được gắn tag “**RELEASE_x**” (x là số hoặc mã của deadline tương ứng).
- Sử dụng GitHub Issues để quản lý các vấn đề trong quá trình phát triển và GitHub **Wiki để mô tả chi tiết mọi thành phần của đồ án**: Đặc tả chức năng, các bản thiết kế, đặc tả chi tiết từng module trong code.
- Cập nhật liên tục, có sự tham gia của tất cả các thành viên trong mã nguồn, issue và wiki.