# IDE

**Môi trường phát triển tích hợp** (tiếng Anh: *Integrated Development Environment*; viết tắt: *IDE*) còn được gọi là "*Môi trường thiết kế hợp nhất*" (tiếng Anh: *Integrated Design Environment*) hay "*Môi trường gỡ lỗi hợp nhất*" (tiếng Anh: *Integrated Debugging Environment*) là một loại phần mềm máy tính có công dụng giúp đỡ các lập trình viên trong việc phát triển phần mềm.

Các môi trường phát triển hợp nhất thường bao gồm:

* Một trình soạn thảo mã (*source code editor*): dùng để viết mã.
* [Trình biên dịch](http://vi.wikipedia.org/wiki/Tr%C3%ACnh_bi%C3%AAn_d%E1%BB%8Bch) (*compiler*) và/hoặc trình thông dịch (*interpreter*).
* Công cụ xây dựng tự động: khi sử dụng sẽ biên dịch (hoặc thông dịch) mã nguồn, thực hiện liên kết (*linking*), và có thể chạy chương trình một cách tự động.
* Trình gỡ lỗi (*debugger*): hỗ trợ dò tìm lỗi.
* Ngoài ra, còn có thể bao gồm hệ thống quản lí phiên bản và các công cụ nhằm đơn giản hóa công việc xây dựng giao diện người dùng đồ họa (GUI).
* Nhiều môi trường phát triển hợp nhất hiện đại còn tích hợp trình duyệt lớp (*class browser*), trình quản lí đối tượng (*object inspector*), lược đồ phân cấp lớp (*class hierarchy diagram*),... để sử dụng trong việc phát triển phần mềm theo hướng đối tượng.

Phân theo số lượng các ngôn ngữ được hỗ trợ, ta có thể chia các môi trường phát triển hợp nhất được sử dụng rộng rãi ngày nay thành hai loại:

* Môi trường phát triển hợp nhất một ngôn ngữ: làm việc với một ngôn ngữ cụ thể, ví dụ: Microsoft Visual Basic 6.0 IDE.
* Môi trường phát triển hợp nhất nhiều ngôn ngữ: có thể làm việc với nhiều ngôn ngữ lập trình, ví dụ: Eclipse IDE, NetBeans, Microsoft Visual Studio.

Trong thời gian gần đây, người ta thấy nổi lên các môi trường phát triển hợp nhất nguồn mở (Open Source IDE), như NetBeans, Eclipse. Các môi trường phát triển loại này ngày càng thông dụng, nhất là trong cộng đồng nguồn mở

# Tính năng cần có

|  |  |
| --- | --- |
| Tính năng | Giải thích |
| Hierachy | Trình bày phân cấp các công việc của Project và TFS hổ trợ thể hiện các công việc, các work item được phân cấp, liên kết với nhau như thế nào |
| Successor/Predecessor | Định nghĩa các kiểu lien kết, định nghĩa một kiểu liên kết trình bày tương ứng với mối quan hệ giữa các công việc liên quan vói nhau, việc tiền nhiệm/ việc kế nhiệm |
| Rollups | Làm việc trên các tổng thể thiết kế của các công việc dự án và tính toán để dự án được tốt hơn |
| Undo |  |
| Check-in và check-out | Mỗi khi check-in hệ thống tạo ra một change set (đối tượng quản lý tập các thay đổi về mã nguồn dữ án).Khi check-in một file lên server, có thể có sự xung đột giữa nội dung bản nằm trên server và nội dung khi mới check-out (lấy từ server xuống) trong trường hợp sử dụng tùy chọn multi-check-out và có nhiều lập trình viên cùng sửa một file. Tính năng “Conflict Channel” |
| Branch và Merge | Branch (phân nhánh) dùng để phân nhánh tạo ra phiên bản mới. Ngược lại Merge (hợp) dùng để kết hợp hai phiên bản với nhau. |
| Label | Dùng để đánh dấu, đặt tên một phiên bản ở một thời điểm. Label rất hữu dụng khi đội phát triển cần tìm lại tài nguyên trong quá khứ phát triển. |
| Wiki | Dùng để ghi lại những tin tức quan trọng, các hướng dẫn cơ bản |
| Team Manager | Thay đổi quyền thành viên, thêm thành viên mới |
| Alerts | Thông báo các hoạt động của nhóm |
| Trac/SVN | quản lý code source, project management và bug-tracking |
| Milestones | Lập kế hoạch và công việc cụ thể |