



Introduction to developing on MindTouch

This document is in-progress and is not complete.

This document is a short primer on the MindTouch technology stack to give interested developers an idea of best development practices.

Goals

As a platform, the primary goal of developers building on top of MindTouch is to (1) minimize time of development, (2) follow best practices to avoid scalability/security hole concerns and (3) ensure all work down is future-proof in terms of upgrades. MindTouch's platform is quite extensible, and has many integration points - this document will help refine decisions with regards to specific integration points.

While MindTouch cannot guarantee 100% backwards compatibility on the platform, our primary goal with every release is to introduce no breaking changes on publicly stable APIs. MindTouch makes no guarantees on any undocumented or experimental tools.

Platform Overview

The best place to understand our architecture is to the [MindTouch technology page](#). The short version is that MindTouch is essentially tiered into a PHP front-end, and the API C# back-end (.NET/Mono). If you're doing UI work (includes skinning and any customizations of the UI workflow), you will most likely be working with [hooks & plugins](#) (PHP) or our [skinning engine](#) (PHP). For data integration, you will be working with [MindTouch Extensions](#) or [DekiScript](#). For mashups (both external data sources and internal MindTouch data), you will primarily utilize [DekiScript](#).

To extend MindTouch's API, it's possible to write your own C# services using [MindTouch Extensions](#) - as an example, we've done this with our [drafts implementation](#).

Developer Classes

Designers

We love standards-driven designs. For designers, an understanding of XHTML/CSS is critical; an ability to understand our PHP templating engine (*hardcore PHP experience is not required!*) will go a long way to making beautiful skins. (jQuery is also included inside the core MindTouch app - *move this*) Learn more in our [skinning documentation](#).

UI Developer

For the class of developers who have more PHP knowledge, our hooks & plug-ins allow you to prioritize custom PHP code ahead of ours, essentially allowing you to bypass any of our workflows; this is common methodology utilized in other PHP apps like Drupal and WordPress. This is useful if you want to do things like hook on additional behavior after log-in, do post-processing on page saves, and more. Learn more in our [hooks and plugins](#) documentation.

For UI development inside MindTouch pages, there is an experimental library called **JEM** which allows you to write a hybrid of Javascript and DekiScript to build UIs within DekiScript.

Editor customization

FCKeditor has its own methods for allowing custom editor plug-ins. Through our control panel, MindTouch administrators can set their own FCKeditor configuration. (*Needs appropriate MindTouch page*)

Developers

Developing new core features inside MindTouch (outside of the extension realm, which is language-agnostic) will require either PHP or C# experience. For any features that require writing significant amounts of data, it is suggested you write your feature as an API service (database access from the front-end is being deprecated to ease scalability concerns).

For C# developers, you'll need to first understand **MindTouch Dream** and how to make your application an extension (*service?*).

