You have to rely on the iOS that the MFMailComposeResult that is handed back in mailComposeController:didFinishWithResult:error: is correct. The simulator fakes that result; no actual mail is sent although it says MFMailComposeResultSent.

**The tutorial mentioned misses an important point**: The first thing you should do before using MFMailComposeViewController is to check [MFMailComposeViewController canSendMail]. That will return NO, if the user hasn't configured mail on their device. If you must support an iOS version prior to 3.0 the correct way is to check if the class MFMailComposeViewController exists:

Class mailClass = (NSClassFromString(@"MFMailComposeViewController")); if (mailClass != nil) { if ([mailClass canSendMail]) { [self displayComposerSheet]; } else { [self launchMailAppOnDevice]; } } else { [self launchMailAppOnDevice]; }

The canSendMail-issue can only be tested on a real device though. It will crash if you don't check canSendMail and the user has no mail account configured.