Composer Web Installer

As a follow-up to Composer Dependencies Revamp, let's plan how to make it easy for site admins to manage Composer dependencies via the web interface. This will be experimental for Tiki17 and stable for Tiki18.

For now, we focus on packages from https://packagist.org/ but this will later be extended to

- themes
- eventually turn off mods https://mods.tiki.org/, which is only used currently for PluginR to get around licensing issues (GPL vs LGPL).
- and perhaps translations.

So perhaps Preferences should have a new concept of external dependency?

Site admin experience

1. Can test that everything is OK: https://getcomposer.org/doc/03-cli.md#diagnose
2. Activates a feature which depends on a lib that is not installed (ex.: mPDF)
3. Graceful error message explaining what dependency is missing and indicates the license
   ○ If server permits, site admin is offered a link / button to install the dependency
     ▪ Question: What do we do for versionning? Ex.: Do we let site admins install a Beta version? Ex. mPDF 7.x is in Beta. Do we want to promote? We do want to only promote versions that have explicitly tested by a community member. They could be suggested in the composer.json.dist? https://getcomposer.org/doc/04-schema.md#suggest And visible in the GUI somehow?
     ▪ If server doesn't permit, indicate what setting to change
   ○ Manual instructions are also supplied (shell command or get this zip, and unzip in a specific directory)
     ▪ Perhaps using the Phar format?
4. System checks all worked well and reports success or failure.
5. Ideally, secDB is updated (low priority): All Tiki files have a feature and permission check which makes them only a risk if active but we don't have this for external libs.

- If vendor/ is later no longer there (purposely deleted or database is pointed to a fresh check-out of Tiki), a warning should appear on the top of tiki-admin.php

First packages to get working

- mPDF
  ○ cp composer.json.dist composer.json
  ○ php temp/composer.phar require mpdf/mpdf:^6.1

- PhantomJS and CasperJS (will be interesting to test on Windows)
- ViewerJS but we have a problem:
Exposé IDS
php-saml

Priority is GNU/Linux but later Windows and OSX should be tested as well