

February 22, 2014

# MobileMaker Design Specification

## *Version 1.0*

Prepared for CS189A  
Chandra Krintz  
Geoffrey Douglas

### MobileMaker Members

Andres Riofrio <riofrios@gmail.com>  
Daniel Retta <dretta@ymail.com>  
David Carlson <davidcarlson2014@gmail.com>  
Marco Sedano <MarcoASedano@gmail.com>  
Nicholas Marks <nickloop@gmail.com>

# Table of Contents

## [Table of Contents](#)

### [1. Introduction](#)

#### [1.1 Product Overview](#)

#### [1.2 Definitions, Acronyms, Abbreviations](#)

#### [1.3 Document Overview](#)

### [2. Components](#)

#### [2.1 Command Line Interface](#)

#### [2.2 Graphical User Interface](#)

#### [2.3 Magento Toolkit](#)

#### [2.4 MySQL Database](#)

### [3. Design Specifications](#)

#### [3.1 High Level Overview](#)

#### [3.2 Automatically Configure Mobile Site](#)

##### [3.2.1 Create a New Store View](#)

###### [3.2.1.1 There does not exist a mobile store view with that name](#)

###### [3.2.1.2 There already exists a mobile store view with that name](#)

##### [3.2.2 Create Design Change](#)

###### [3.2.2.1 There is not a design selected to change](#)

###### [3.2.2.2 There is a design selected to change](#)

##### [3.2.3 Add Mobile Detection Code to “.htaccess” file](#)

###### [3.2.3.1 “.htaccess” file is not found](#)

###### [3.2.3.2 “.htaccess” file is found and mobile detection code \[TODO!\]](#)

#### [3.3 Automatically Delete Mobile Site](#)

##### [3.3.1 Remove the Store View and Design Change](#)

###### [3.3.1.1 The Store View and Design Change exist](#)

###### [3.3.1.2 The Store View and Design Change do not exist](#)

#### [3.4 Get overall color scheme from the desktop site](#)

##### [3.4.1 Get Color Scheme](#)

##### [3.4.2 List Color Scheme](#)

###### [3.4.2.1 Command Line](#)

###### [3.4.2.2 Graphical](#)

#### [3.5 Allow user to change color, font, image, etc.](#)

##### [Three less files which are related to the iPhone and custom css files.](#)

##### [3.5.1 Create default settings.less and change the variables to control Color, Font, Image, etc.](#)

##### [3.5.2 Open settings.less and change the variables to control Color, Font, Image, etc.](#)

#### [3.6 Enable and disable the mobile site](#)

##### [3.6.1 Enable command](#)

#### [3.7 Change the theme\\*](#)

#### [3.8 Customize the colors](#)

[There are three stylesheets in customizable themes:](#)

[4. Class Diagram](#)

[5. UI Mockups](#)

[5.1 Command Line Interface](#)

[5.1.1 Usage \(no arguments\)](#)

[5.1.2 Help command](#)

[5.1.3 Init command](#)

[5.1.4 Enable command](#)

[5.1.4.2 Mobile Site is running and .htaccess file is enabled](#)

[5.1.4.3 Mobile Site is running and detection codes is disabled](#)

[5.1.6 Destroy command](#)

[5.1.6.2 Mobile Site is running and .htaccess file is enabled](#)

[Glossary](#)

# 1. Introduction

## 1.1 Product Overview

E-commerce is a great way for companies to sell large amounts of products without physically having stores in all the areas where a product is needed. By ordering online the customer can quickly and easily purchase items and have them delivered to their door. Many companies use an e-commerce tool called Magento to manage their website and store database but currently do not have a website formatted for mobile. In order for companies to have a sleek mobile site they must write the code themselves. Warp 9 has challenged five University of California, Santa Barbara students with the task of automating the initial steps of creating a custom mobile site. Mobile Maker is our project which attempts to solve this problem.

## 1.2 Document Overview

Section 2 consists of a very high level summary of the components of the system. Section 3 consists of detailed descriptions of each feature: mobile site creation, enabling and disabling, customizing the theme, etc. Section 4 contains a class diagram. Finally, Section 5 contains mockups of the Command Line Interface.

## 2. Components

### 2.1 Command Line Interface

A well documented CLI toolkit which enables users to quickly and efficiently download, initialize, and customize mobile templates associated with a Magento Store. The Command Line Interface will have many options to give the user maximum control over the process of customizing their mobile site as well as giving novice users defaults which set up a sleek mobile site.

### 2.2 Graphical User Interface

The GUI will allow novice users and non-programmers to perform the same tasks our CLI can perform, but with more visual feedback. The actions that can be performed with the GUI will correspond to the actions the CLI can perform, and they will share the appropriate code.

### 2.3 Magento Toolkit

Magento is an e-commerce platform developed by eBay. Magento is a content management system for e-commerce, and can be customized almost limitlessly through configuration settings (through the GUI), and through themes and extensions (code).

### 2.4 MySQL Database

The MySQL database is a data repository that is used by Magento to store the store items, as well as the configuration values that control how the web store is displayed and how it behaves. Not all configuration is stored in the database. Some of it is stored as XML files in the filesystem. When we need to access the database to perform the tasks our CLI or GUI require, we will do so using the Magento ORM (Object Relational Mapping) APIs.

### 3. Design Specifications

We focused on designing a command line Tool that Warp 9 developers can use to more quickly setup a mobile site for their clients. This command line tool is called “mobilize.php” and contains multiple subcommands. For example, to run the “init” command, one would type:

```
php mobilize.php init [OPTIONS]
```

For our reference, features that have not been implemented yet are marked with an asterisk (\*).

#### 3.1 High Level Overview

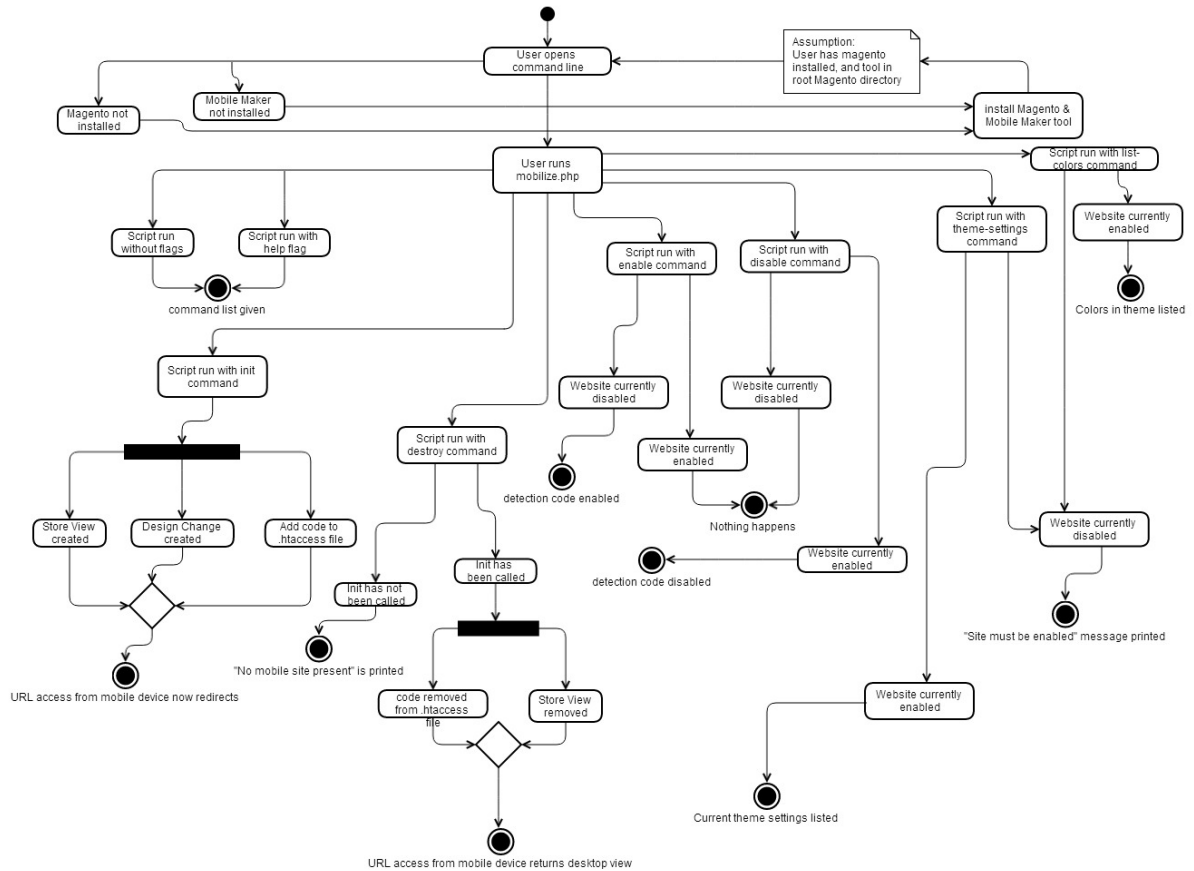


Figure 3.1: The figure above shows a high level expected interaction between the user and Magento web application the user is expected to run MobileMaker.

## 3.2 Automatically Configure Mobile Site

The “init” command creates a mobile store view associated with the code “mobile”, adds a design change that will connect the mobile store view to the default/iphone-mobilemaker mobile theme, and finally adds a few lines of code to the .htaccess file which alerts the system whether it is communicating with a mobile or a desktop web browser.

The option --theme=THEME allows the user to specify a Magento theme to use instead of the default/iphone-mobilemaker theme.\*

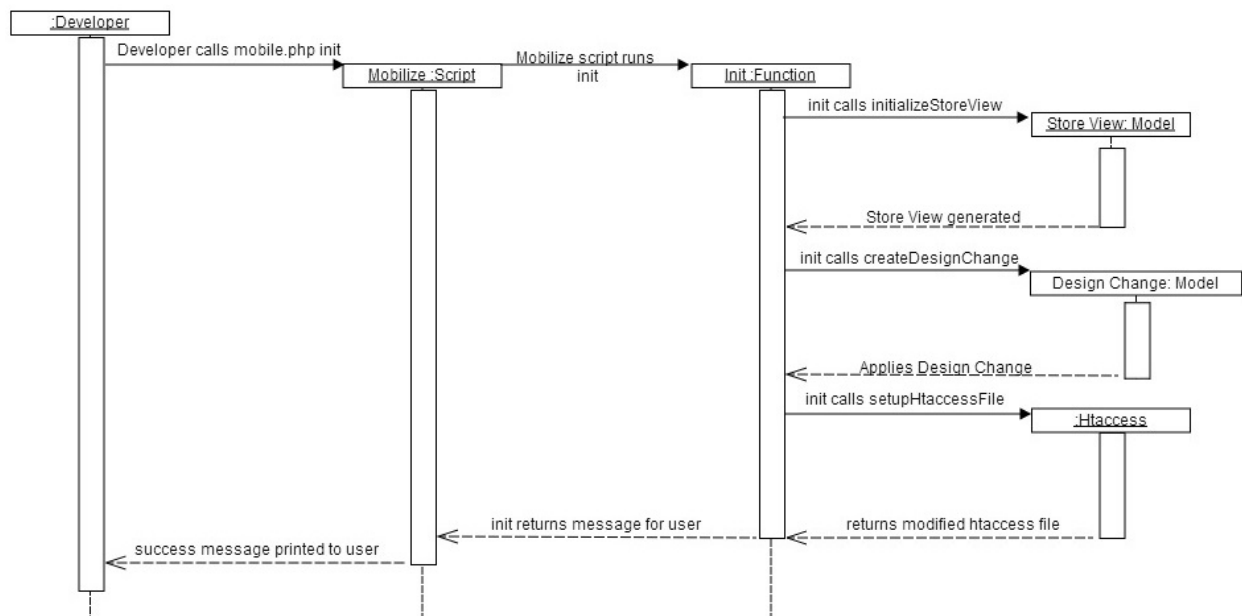


Figure 3.2: The figure above shows an overview of the init command’s functionality in mobilize.php

### 3.2.1 Create a New Store View

#### 3.2.1.1 There does not exist a mobile store view with that name

A new store view will be made with mobile store code “mobile”.

#### 3.2.1.2 There already exists a mobile store view with that name

Alert the user that a mobile store already exists with the name specified and the user should enter a unique ID for the store view.

## 3.2.2 Create Design Change

### 3.2.2.1 There is not a design selected to change

The user will be prompted to select a store design of their choice to change.

### 3.2.2.2 There is a design selected to change

The user will be taken to a new page (or some other entity) where their chosen design has been loaded on. From there they will be able to make edits on the features in the page (or entity).

## 3.2.3 Add Mobile Detection Code to “.htaccess” file

The mobile detection code consists of a few Rewrite rules that detect a mobile device by matching on the User Agent, and then sets the MAGE\_RUN\_CODE and MAGE\_RUN\_TYPE environment variables to instruct Magento to display the mobile site instead of the regular desktop site.

### 3.2.3.1 “.htaccess” file is not found

A new “.htaccess” file will be created with the mobile detection code.

### 3.2.3.2 “.htaccess” file is found and mobile detection code is found within the file

An error message is shown.

### 3.2.3.3 “.htaccess” file is found and mobile detection code is *not* found within the file

The mobile detection code is appended to the end of the file.

## 3.2.4 Install customizable theme and required extensions\*

Finally, the “init” command uses Magento Connect (an Magento extension marketplace and package manager) to install the “iphone-mobilemaker” theme and the “magento-lesscss” extension (which adds support for Less to Magento). Less is a CSS preprocessor and stylesheet language that extends CSS with variables and functions. We use variables to allow Warp 9 developers to easily change the color of a group of components by changing the value of the corresponding variable. More details can be found in 3.5.

If the extension is already installed, it will not try to update it. If the theme is already installed, it will not try to update it.

## 3.3 Automatically Delete Mobile Site

The “destroy” command deletes the store view associated with the code “mobile” (the same that is created by the “init” command). Any design changes associated with the store view are automatically deleted by Magento. Then, it removes the mobile detection code from the .htaccess file in the same way as if the “disable” command was run.



This command first calls `Mage::register()` to set the “isSecureArea” configuration variable to true. This tells Magento to enable potentially destructive features as if the administrator was logged in, thus allowing us to delete the store view.

### 3.3.1 Remove the Store View and Design Change

#### 3.3.1.1 The Store View exists

Mobile Store View and design changes are deleted. Mobile detection code is removed.

#### 3.3.1.2 The Store View does not exist

Program alerts the user that there is no existing store view and therefore nothing to delete. Mobile detection code is not touched (but it should not be present or the site will show a 404 error when accessed from a mobile device).

## 3.4 Get overall color scheme from the desktop site

The “list-colors” command lists every color used in the default store view’s theme (by default, this is “default/default”).

To get every color used in a specific theme in Magento, you must search through the directory containing all of the theme’s CSS files. Once you locate this directory (in the “magento/skin/\$PACKAGE/\$THEME” directory) you can traverse each file and record each unique color found. These colors can be found by searching for strings beginning with a pound sign and followed by 3 or 6 hexadecimal values. These hexadecimal values correspond to a specific RGB value. This search can be accomplished by utilizing the following regular expression:

```
/#((([a-f]|[A-F]|[0-9]){6})|(([a-f]|[A-F]|[0-9]){3}))/
```

## 3.5 Customize the colors\*

The “customize” command opens a file that the user can use to customize the colors of the theme. The editor to use is chosen according to the following algorithm:

First, try `ENV['EDITOR']`. If that fails, try “vi”. If that fails, just print the path to the file explaining the problem and asking them to open the file in a text editor to modify the theme settings.

If the mobile site is not setup, or if the theme is not customizable, an error message is shown.

There are three stylesheets in customizable themes:

- mobilemaker.less is a copy of custom.css which has variables in place for anything the user would want to change.
- defaults.less holds default values for the less variables which are found in the original theme.
- user.less is used by the user to overwrite the variables. This is the file that is opened by the “customize” command.

### **3.6 Enable and disable the mobile site**

The “enable” command adds the mobile detection code to the “.htaccess” file. The “disable” command removes this code from the “.htaccess” file.

#### **3.6.1 Enable command**

If the mobile site has not been created (with the “init” command), or if the site has already been enabled (the mobile detection code is found in the “.htaccess” file), an error message is shown. Otherwise, the mobile detection code is added to the “.htaccess” file.

#### **3.6.2 Disable command**

If the mobile site has not been created (with the “init” command), or if the site has already been disabled (the mobile detection code is not found in the “.htaccess” file), an error message is shown. Otherwise, the mobile detection code is removed from the “.htaccess” file.

### **3.7 Change the theme\***

The “change-theme” command modifies the Design Change associated with the mobile site to refer to another theme. It takes an argument, which is the name of the Magento theme, which will be downloaded and installed from Magento Connect if necessary.

Any theme can be enabled using this method, but customization of colors will only work on supported themes that we build. We will begin with one theme called default/iphone-mobilemaker which is based on default/iphone.

## 4. Class Diagram

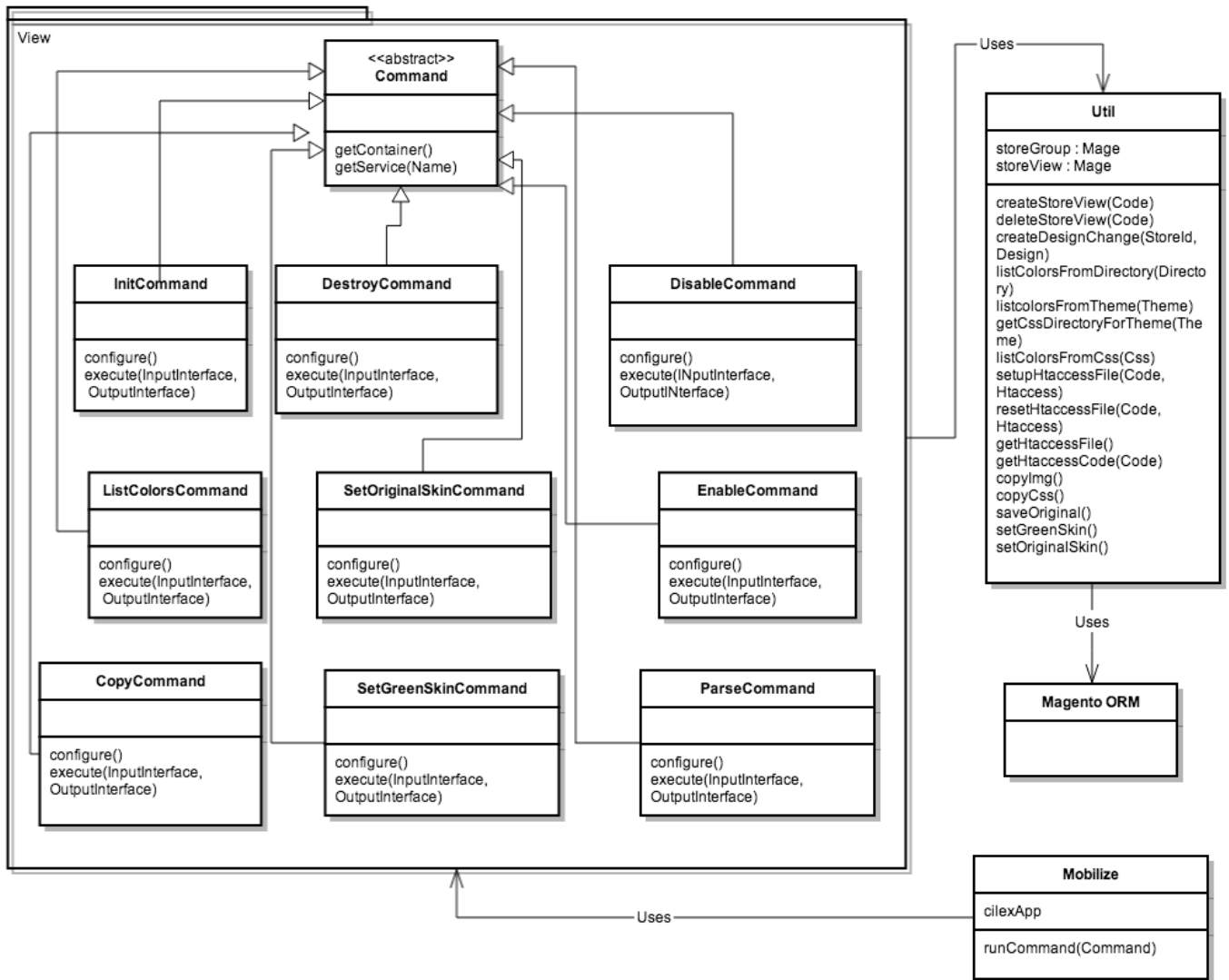


Figure 4.0: Here we have the class diagram for our Mobile Maker. All of the commands inherit from the abstract class **Command**, while **Mobilize** Uses all of the commands when running **Mobilize**.

## 5. UI Mockups

### 5.1 Command Line Interface

#### 5.1.1 Usage (no arguments)

```
$ php mobilize.php  
Mobilize version 1.0
```

Usage:

```
[options] command [arguments]
```

Options:

```
--help          -h Display this help message.  
--quiet         -q Do not output any message.  
--verbose       -v|vv|vvv Increase the verbosity of messages: 1 for normal  
output, 2 for more verbose output and 3 for debug  
--version       -V Display this application version.  
--ansi          Force ANSI output.  
--no-ansi       Disable ANSI output.  
--no-interaction -n Do not ask any interactive question.
```

Available commands:

```
help           Display this message  
list           Lists commands  
init           Add a mobile site and enable it for mobile devices  
destroy        Disable and remove the mobile site  
enable         Enable the mobile site by adding detection code to the  
               htaccess file  
disable        Disable the mobile site by removing detection code from the  
               htaccess file  
theme-settings Opens the settings.less file  
list-colors    Lists all the colors in the default desktop theme
```

## 5.1.2 Help command

```
$ php mobilize.php help
```

Usage:

```
help [--xml] [--format="..."] [--raw] [command_name]
```

Arguments:

command	The command to execute
command_name	The command name (default: "help")

Options:

--xml	To output help as XML
--format	To output help in other formats (default: "txt")
--raw	To output raw command help
--help (-h)	Display this help message.
--quiet (-q)	Do not output any message.
--verbose (-v:vv:vvv)	Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug
--version (-V)	Display this application version.
--ansi	Force ANSI output.
--no-ansi	Disable ANSI output.
--no-interaction (-n)	Do not ask any interactive question.

Help:

The help command displays help for a given command:

```
php mobilize.php help list
```

You can also output the help in other formats by using the --format option:

```
php mobilize.php help --format=xml list
```

To display the list of available commands, please use the list command.

## 5.1.3 Init command

### 5.1.3.1 Mobile Site is not running

```
$php mobilize.php init [--theme=THEME]
Creating store view...
Creating design change...
Setting up the .htaccess file..
Done.
```

### 5.1.3.2 Mobile Site is running

```
$php mobilize.php init [--theme THEME]
Creating store view...
```

```
[Mage_Core_Exception]
Store with the same code already exists.
```

```
Done.
```

## 5.1.4 Enable command

### 5.1.4.1 Mobile Site not running\*

```
$php mobilize.php enable
Mobile site must be running before enabling it.
```

### 5.1.4.2 Mobile Site is running and .htaccess file is enabled

```
$php mobilize.php enable
The mobile detection code was already in the .htaccess file.
Done.
```

### 5.1.4.3 Mobile Site is running and detection codes is disabled

```
$php mobilize.php enable
Done.
```

## 5.1.5 Disable command

### 5.1.5.1 Mobile Site not running\*

```
$php mobilize.php enable  
Mobile site must be running before disabling it.
```

### 5.1.5.2 Mobile Site is running and .htaccess file is enabled

```
$php mobilize.php disable  
Done.
```

### 5.1.5.3 Mobile Site is running and detection codes is disabled

```
$php mobilize.php disable  
The mobile detection code was not found in the .htaccess file.  
Done.
```

## 5.1.6 Destroy command

### 5.1.6.1 Mobile Site is not running\*

```
$php mobilize.php destroy  
No mobile site is present to destroy.  
Done.
```

### 5.1.6.2 Mobile Site is running and .htaccess file is enabled

```
$php mobilize.php destroy  
Removing detection code from the .htaccess file...  
Removing store view...  
Done.
```

### 5.1.6.3 Mobile Site is running and detection codes is disabled

```
$php mobilize.php destroy  
Removing detection code from the .htaccess file...  
The mobile detection code was not found in the .htaccess file.  
Removing store view...  
Done.
```

## 5.1.7 Theme Settings command\*

### 5.1.6.1 Mobile Site is not running\*

```
$php mobilize.php theme-settings
```

No mobile store available to view theme settings from.

### 5.1.6.2 Mobile Site is running and .htaccess file is enabled\*

```
$php mobilize.php theme-settings
```

Theme Settings:

...

### 5.1.6.3 Mobile Site is running and detection codes is disabled\*

```
$php mobilize.php theme-settings
```

Mobile store must be enabled to view theme settings.

## 5.1.8 List Colors command

### 5.1.6.1 Mobile Site is not running\*

```
$php mobilize.php list-colors
```

No mobile store available to retrieve colors from.

### 5.1.6.2 Mobile Site is running and .htaccess file is enabled

```
$php mobilize.php list-colors
```

Listing colors...

Here are the colors:

...

### 5.1.6.3 Mobile Site is running and detection codes is disabled\*

```
$php mobilize.php list-colors
```

Mobile store must be enabled to retrieved colors.