

Messenger Methods

Table of Contents

- 1 Required Methods
 - 1.1 getName()
 - 1.2 getVersion()
 - 1.3 getAuthors()
 - 1.4 getTypes()
 - 1.5 send(\$to_user_id, \$content, \$type = null)
- 2 Optional Methods
 - 2.1 install/upgrade/uninstall()
 - 2.2 getDescription()
 - 2.3 getLogo()

Required Methods



All of these required methods can be defined in a [messenger configuration file](#) instead.

The following methods are required to be implemented for each messenger.

getName()

The `getName()` method simply returns the name of the messenger. It's always best to define any language in your messenger using language files (see [Translating Blesta](#) for more information).

```
class MyMessenger extends Messenger {
    ...

    public function getName()
    {
        return Language::_('MyMessenger.name', true);
    }
}
```

getVersion()

This method must return the current version number of the messenger. Upon installation or upgrade, Blesta records the current code version so that it can tell when an upgrade/downgrade is available. The version number should be compatible with PHP's [version_compare\(\)](#) function.

```
class MyMessenger extends Messenger {
    const VERSION = '1.0.0';

    ...

    public function getVersion()
    {
        return self::VERSION;
    }
}
```

getAuthors()

This method returns an array containing information about the authors of the messenger.

```
class MyMessenger extends Messenger {
    private static $authors = [
        ['name' => 'Phillips Data, Inc.', 'url' => 'http://www.blesta.com']
    ];
    ...

    public function getAuthors()
    {
        return self::$authors;
    }
}
```

getTypes()

This method returns a list of the message types supported by the messenger.

```

class MyMessenger extends Messenger {

    ...

    public function getTypes()
    {
        return ['sms'];
    }
}

```

send(\$to_user_id, \$content, \$type = null)

This method sends the message.

```

class MyMessenger extends Messenger {

    ...

    public function send($to_user_id, $content, $type = null)
    {
        Loader::loadModels($this, ['Staff', 'Clients']);

        // Fetch user information
        $is_client = true;
        if (($user = $this->Staff->getByUserId($to_user_id))) {
            $is_client = false;
        } else {
            $user = $this->Clients->getByUserId($to_user_id);
        }

        // Send message
        $error = null;
        $success = false;

        if ($type == 'sms') {
            ...

            $success = true;
        }

        $this->log($to_user_id, $content, $error, $success);
    }
}

```

Optional Methods

The methods below are optional.

install/upgrade/uninstall()

The methods are invoked when the messenger is installed, upgraded, or uninstalled respectively.

```

class MyMessenger extends Messenger {

    ...

    public function install()
    {
        // Perform install logic
    }

    public function upgrade($current_version)
    {
        // Perform upgrade logic
    }

    public function uninstall($messenger_id, $last_instance)
    {
        // Perform uninstall logic
    }
}

```

getDescription()

The getDescription() method simply returns the description of the messenger. It's always best to define any language in your messenger using language files (see [Translating Blesta](#) for more information).

```

class MyMessenger extends Messenger {

    ...

    public function getDescription()
    {
        return Language::_('MyMessenger.description', true);
    }
}

```

getLogo()

The getLogo() method returns the relative path (within the messenger's directory) to the logo used to represent the messenger. The default value is **views/default/images/logo.png**. This translates to **/install_dir/components/messengers/my_messenger/views/default/images/logo.png**.

```

class MyMessenger extends Messenger {

    ...

    public function getLogo()
    {
        return 'some/path/to/my/logo.png';
    }
}

```