

---

# **atilla documentation**

***Release 1.2.6***

**Oleg Pidsadnyi, Paylogic International and others**

**Mar 13, 2017**



---

## Contents

---

<b>1</b>	<b>Atila</b>	<b>3</b>
1.1	Installation . . . . .	3
1.2	Documentation . . . . .	3
1.3	Contact . . . . .	3
1.4	License . . . . .	3
<b>2</b>	<b>Internal API</b>	<b>5</b>
	<b>Python Module Index</b>	<b>7</b>



## Contents

- *Welcome to atilla's documentation!*
- *Atilla*
  - *Installation*
  - *Documentation*
  - *Contact*
  - *License*
- *Internal API*



# CHAPTER 1

---

## Atila

---

Atila (the API Utilities Library) contains a number of useful functions that are common between flask projects for HTTP API applications. This includes types, validators, functions for creating applications, etc. This library will make setting up new projects a lot faster.

## Installation

```
pip install atilla
```

## Documentation

Detailed python API you can find in the [documentation](#).

## Contact

If you have questions, bug reports, suggestions, etc. please create an issue on the [GitHub project page](#).

## License

This software is licensed under the [MIT license](#)

See [License file](#)

© 2015 Paylogic International and others.





Database transaction related functionality.

**class** `atilla.transaction.DatabaseManager` (*session*, \**args*, \*\**kwargs*)  
Database manager and thread-safe settings.

**close** ()  
Close all database sessions.

**commit** ()  
Commit and close the session.

**Note** Rollback the session on error.

**Raise** The exception is re-raised.

**rollback** ()  
Rollback all database sessions.

`atilla.transaction.create_db_manager` (*app*, *session*)  
Create db manager and assign it on app.

**Parameters** **session** – *Session* object



### **a**

`atilla.transaction`, 5



### A

`atilla.transaction` (module), [5](#)

### C

`close()` (`atilla.transaction.DatabaseManager` method), [5](#)

`commit()` (`atilla.transaction.DatabaseManager` method), [5](#)

`create_db_manager()` (in module `atilla.transaction`), [5](#)

### D

`DatabaseManager` (class in `atilla.transaction`), [5](#)

### R

`rollback()` (`atilla.transaction.DatabaseManager` method),  
[5](#)