

Model

User
<div>- dataTable: 'user'</div> <div>- fname: string</div> <div>- lname: string</div> <div>- email: string</div> <div>- password: string</div> <div>- user_uuid: string</div>
<div>+ getUserId(): object</div> <div>+ getEmail(object): string</div> <div>+ authenticate(string, string, array): bool</div> <div>+ registerUser(string, string, string, string, string):bool</div> <div>- confirmEmail(string, string): bool</div> <div>- confirmPassword(string, string): bool</div> <div>- saltPassword(string): string</div> <div># createUserSession(object): void</div> <div>+ destroyUserSession(): void</div>

Post
<div>- dataTable: 'posts'</div> <div>- body: string</div> <div>- title: string</div> <div>- post_uuid: string</div>
<div>- getUserId(): string</div> <div>+ getPosts(): array</div> <div>+ getPostByPostUuid(string): object</div> <div>+ createPost(string, string): bool</div> <div>+ editPost(string, string, string): bool</div> <div>+ deletePost(string): bool</div> <div>- checkFields(string, string): bool</div>

View

Includes

<<Interface>>  
Header

- <head>  
+ bootstrap  
+ fontawesome:

<<Interface>>  
Footer

- <script>:jquery  
- <script>:cloudfare  
- <script>:bootstrap

<<Interface>>  
Navbar

+ <navbar>:view

- isLoggedIn():bool  
- getField(string):string  
+ logoutPath(string):controller  
+ loginPath(string):controller  
+ registerPath(string):controller

Pages

<b>&lt;&lt;Interface&gt;&gt; Not_Found</b>
+ loginPath(string):controller

**Posts**

<b>&lt;&lt;Interface&gt;&gt; Index</b>
+ fname:string + lname:string + dateCreated:date + title:string + title:body
+ foreach(array):array + postsEdit(string):controller + postsDelete(string):controller

<b>&lt;&lt;Interface&gt;&gt; Edit</b>
+ postsEdited(array):string + title(array):string + body(array):string

<b>&lt;&lt;Interface&gt;&gt; Delete_Error</b>
+ postsHome(string):controller

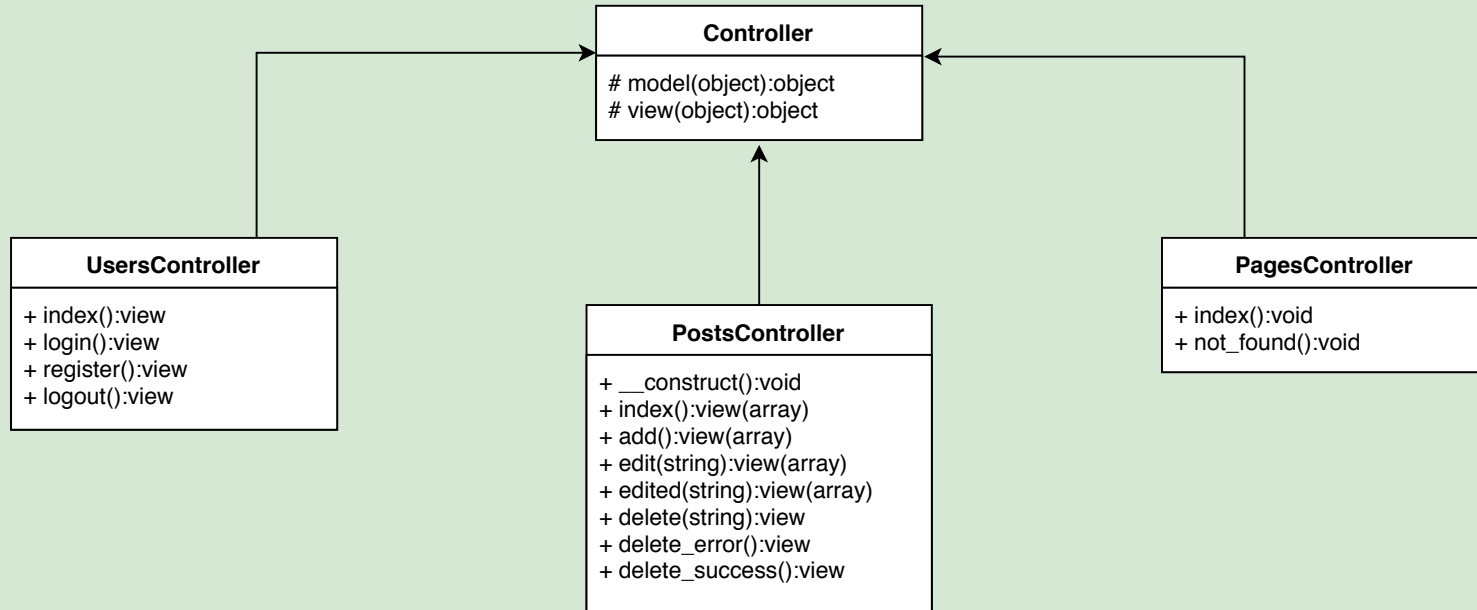
<b>&lt;&lt;Interface&gt;&gt; Delete_Success</b>
+ postsHome(string):controller

**Users**

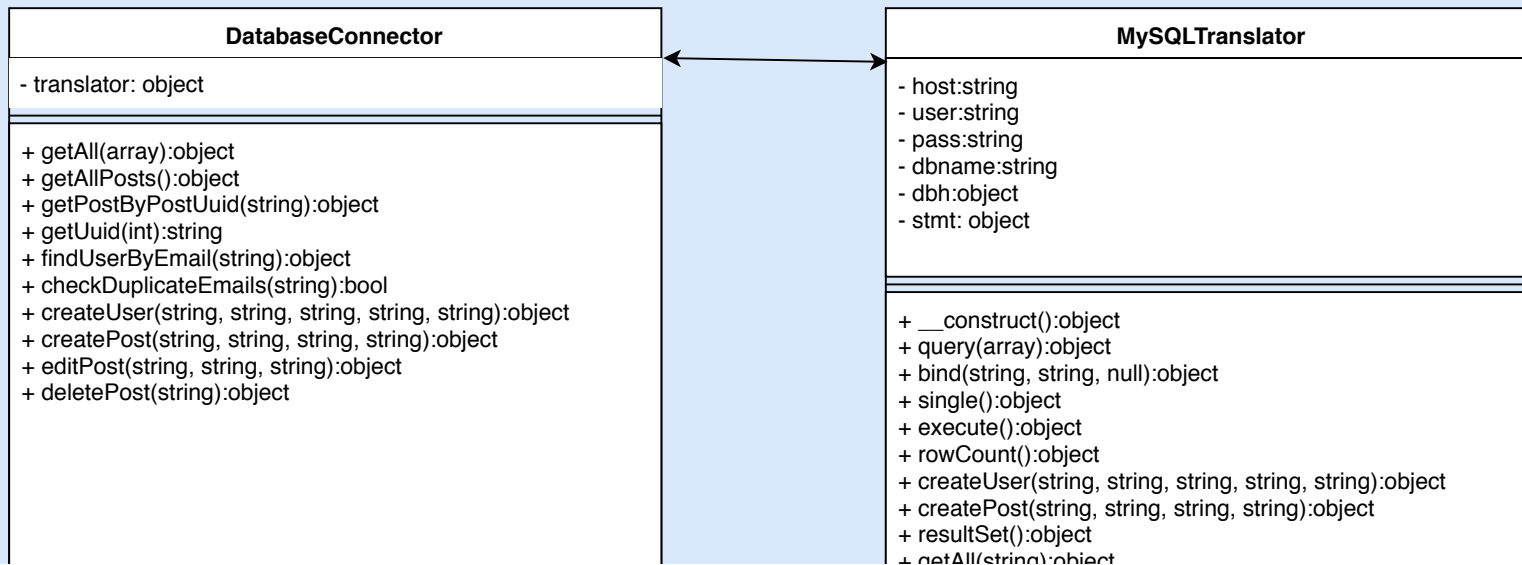
<b>&lt;&lt;Interface&gt;&gt; Index</b>
+ form:array + email:string + password:string
+ button(array):controller + registerPath(string):controller

<b>&lt;&lt;Interface&gt;&gt; Register</b>
+ form:array + fname:string + lname:string + email:string + confirmPassword:string + password:string + confirmPassword:string
+ button(array):controller + registerPath(string):controller

## Controller

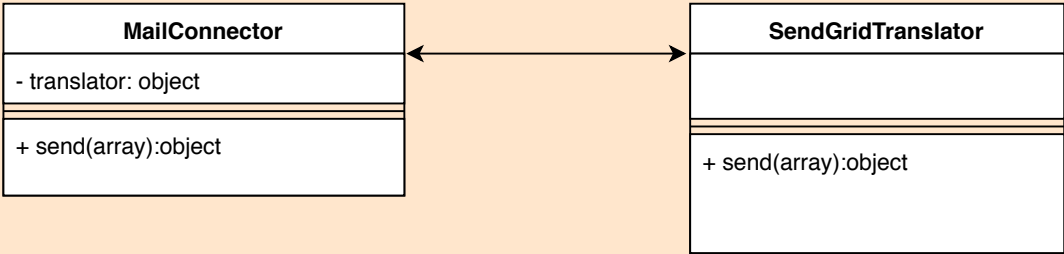


## Data

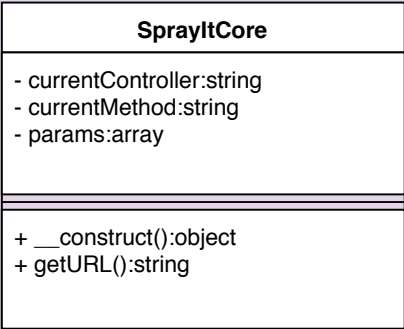
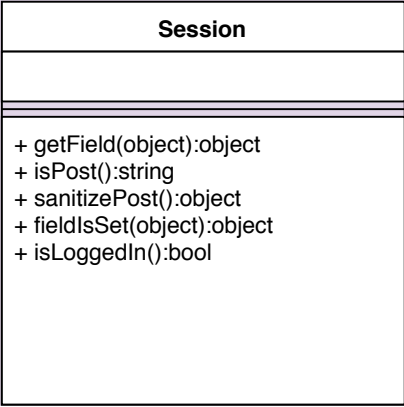
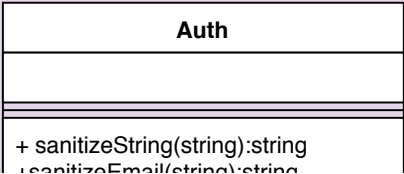
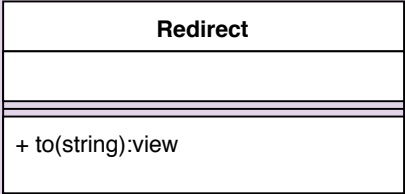


+ getAll(string):object  
+ findUserByEmail(string):object  
+ getAllPosts():object  
+ checkDuplicateEmails(string):bool  
+ getPostByPostUuid(string):object  
+ getUuid(int):string  
+ editPost(string, string, string):object  
+ deletePost(string):object

API



Libraries



+sanitizeEmail(string).string