

Table	Field	Index	Relationships	Type/Constraints
game	event_id	PRI		varchar(255)
	date			datetime
	status			varchar(255)
	away_team_id	MUL	FK to auth_user	varchar(255)
	home_team_id	MUL	FK to auth_user	varchar(255)
	venue_id	MUL	FK to venue	varchar(255)
team	team_id	PRI		varchar(255)
	name			varchar(255)
	wins			int(11)
	losses			int(11)
	venue	UNI	FK to venue (unique enforced)	varchar(255)
venue	venue_id			varchar(255)
	city			varchar(255)
	capacity			int(11)
	name			varchar(255)
	zip			varchar(255)
	country			varchar(255)
	state			varchar(255)
	address			varchar(255)
wagyr	id	PRI		int(11) Auto Increment
	opponent_id	MUL	FK to auth_user	varchar(30)
	amount			decimal(6,2)
	status			int(11)

	self_team			varchar(255)
	opponent_team			varchar(255)
	game_id	MUL	FK to game	varchar(255)
	self_id	MUL	FK to auth_user	varchar(30)
	wagyr_winner	MUL	FK to auth_user	varchar(30)
auth_user	id	PRI		int(11)
	password			varchar(128)
	username	UNI	FK to other auth tables	varchar(30)
	first_name			varchar(30)
	last_name			varchar(30)
	email			varchar(254)
	date_joined			datetime
api_cache_table	cache_key	PRI		varchar(255)
	value			longtext
	expires	MUL	Dump cache after certain time, controlled outside of application	datetime

R/W	Data Source	TestID	Pass/Fail
-	N/A		
R	views.index	data.index.1	P
R	views.search_post	data.search_post.1	P
R	views.index	data.index.2	P
R	views.search_post	data.search_post.2	P
W	forms.SearchGameByTeam.team		P
R	views.index		P
R	views.MakeWagyr	data.index.3	P
R	views.search_post	data.search_post.3	P
W	forms.SearchGameByTeam.team		P
R	views.index		P
R	views.MakeWagyr	data.index.4	P
R	views.search_post	data.search_post.4	P
-	N/A	-	-
R	N/A	-	-
R	views.index	data.index.3,4	P
R	views.search_post	data.search_post.3,4	P
R	views.index	data.index.5	P
R	views.search_post	data.search_post.5	P
R	views.index	data.index.6	P
R	views.search_post	data.search_post.6	P
R	N/A		
-	N/A	-	-
-	N/A	-	-
-	N/A	-	-
R	views.index	data.index.7	P
R	views.search_post	data.search_post.7	P
-	N/A	-	-
-	N/A	-	-
-	N/A	-	-
-	N/A	-	-
-	Auto Incremented (N/A)	-	-
R	views.wagyr	data.wagyr.8	P
W	views.MakeWagyr	data.MakeWagyr.8	P
R	views.wagyr	data.wagyr.9	P
W	views.MakeWagyr	data.MakeWagyr.9	P
R	views.wagyr	data.wagyr.10	P

R	views.wagyr	data.wagyr.11	F
W	views.MakeWagyr	data.MakeWagyr.11	
R	views.wagyr	data.wagyr.12	F
W	views.MakeWagyr	data.MakeWagyr.12	
-	N/A	-	-
-	N/A	-	-
R	views.wagyr	data.wagyr.13	F
-	Auto Incremented (N/A)	-	-
R	forms.LoginForm.password		P
W	forms.UserCreateForm.password1 and		P
W	forms.UserCreateForm.password2		P
W	forms.CrispyPasswordChangeForm.password1 and		P
W	forms.CrispyPasswordChangeForm.password2 and		P
R	forms.CrispyPasswordChangeForm.password	data.user.14	P
R	forms.LoginForm.username		P
W	forms.UserCreateForm.username		P
R	forms.CrispyPasswordChangeForm.username	data.user.15	P
W	forms.UserCreateForm.first_name		P
R	views.profile	data.user.16	
W	forms.UserCreateForm.last_name		P
R	views.profile	data.user.17	
W	forms.UserCreateForm.email		P
R	views.profile	data.user.18	
R	views.profile	data.user.19	P
-	N/A	-	-
-	N/A	-	-
-	N/A	-	-

Notes

Used in internal references. Not exposed.
Only accepts datetime objects. (does not accept durations)

The form is validated by referencing the database. Wrong data never is entered.
Only accepts strings.

The form is validated by referencing the database. Wrong data never is entered.
Only accepts strings.
Used as reference. See Venue.name

Used in internal references. Not exposed.
These columns are used via reference from Game table. See data test 3,4 (row 5,6)

Used in internal references. Not exposed.

Used in internal references. Not exposed.

Used in internal references. Not exposed.
Can select only users that are active in a dropdown
Only able to enter data as decimal value.
Otherwise it throws an error.

Should be foreign key

Should be foreign key

Should only be one of two players in wagyr, not any user.

NOTE: Django controlled, we simply create a foreign key relationship in Wagyr. The forms that fill this table are an extension of Django auth forms, which have been intensively tested by the framework. All validation is handled by the `django.contrib.auth.forms` backend.

This table is used exclusively as a fast cache that supports `get()`, `set()`, `delete()`. It is only used internally to prevent the API call from being necessary.