

Package models

```
import "PGL/APIServer/models"
```

[Overview](#)[Index](#)

Overview ▾

Package model provides the base of the project, which are the working structures

Index ▾

[type Category](#)
[type Inv](#)
[type Item](#)
[type OtherRes](#)
[type User](#)
[type UserSetting](#)

Package files

[models.go](#)

type Category

Category details, UserID->User.ID

```
type Category struct {  
    ID      uint32 `json:"id" gorm:"primaryKey;autoIncrement;not null"`  
    UserID  uint32 `json:"userid" gorm:"not null"`  
    Name    string `json:"name" gorm:"not null"`  
  
    Updated uint64 `json:"updated" gorm:"autoUpdateTime;not null"`  
    Created uint64 `json:"created" gorm:"autoCreateTime;not null"`  
}
```

type Inv

Inventory details, UserID->User.ID

```
type Inv struct {  
    ID      uint32 `json:"id" gorm:"primaryKey;autoIncrement;not null"`  
    UserID  uint32 `json:"userid" gorm:"not null"`
```

```

Icon    string `json:"icon" gorm:"not null"`
Name    string `json:"name" gorm:"not null"`
Items   []Item `json:"items" gorm:"-"`

Updated uint64 `json:"updated" gorm:"autoUpdateTime;not null"`
Created uint64 `json:"created" gorm:"autoCreateTime;not null"`
}

```

type Item

Item details, InvID->Inv.ID

```

type Item struct {
    ID          uint32 `json:"id" gorm:"primaryKey;autoIncrement;not null"`
    InvID       uint32 `json:"invid" gorm:"not null"`
    Icon        string `json:"icon" gorm:"not null"`
    Name        string `json:"name" gorm:"not null"`
    Category    string `json:"category" gorm:"not null"`
    Storage     string `json:"storage" gorm:"not null"`
    Expiry      uint64 `json:"expiry" gorm:"not null" //date of expiry, unix`
    Idle        uint64 `json:"idle" gorm:"not null" //date the item is idle`
    Notify      uint32 `json:"notify" gorm:"not null" //days before to notify for expiry`

    Updated uint64 `json:"updated" gorm:"autoUpdateTime;not null"`
    Created uint64 `json:"created" gorm:"autoCreateTime;not null"`
}

```

type OtherRes

Other response

```

type OtherRes struct {
    Msg string `json:"msg"`
}

```

type User

User details

```

type User struct {
    ID          uint32 `json:"id" gorm:"primaryKey;autoIncrement;not null"`
    Username    string `json:"username" gorm:"unique;not null"`
    Password    string `json:"password" gorm:"not null"`
    Phone       string `json:"phone" gorm:"not null"`
    Email       string `json:"email" gorm:"not null"`
    Inv         []Inv  `json:"inv" gorm:"-"`
    Setting     UserSetting `json:"setting" gorm:"-"`

    Updated uint64 `json:"updated" gorm:"autoUpdateTime;not null"`
}

```

```
Created uint64 `json:"created" gorm:"autoCreateTime;not null"`  
}
```

type UserSetting

User settings, UserID->User.ID

```
type UserSetting struct {  
    ID          uint32 `json:"id" gorm:"primaryKey;autoIncrement;not null"`  
    UserID      uint32 `json:"userid" gorm:"not null"`  
    Type        uint32 `json:"- " gorm:"not null"`  
    NotiTime    string `json:"notitime" gorm:"not null"`  
    NotiSetting uint32 `json:"notisetting" gorm:"not null" //1,2`  
    PrevLogin   string `json:"prevlogin" gorm:"not null"`  
    CurrLogin   string `json:"currlogin" gorm:"not null"`  
  
    Updated uint64 `json:"updated" gorm:"autoUpdateTime;not null"`  
    Created uint64 `json:"created" gorm:"autoCreateTime;not null"`  
}
```

Build version go1.15.7.

Except as [noted](#), the content of this page is licensed under the Creative Commons Attribution 3.0 License, and code is licensed under a [BSD license](#).

[Terms of Service](#) | [Privacy Policy](#)