

GreenForge API

API and SDK Documentation

Version: 1.0.0

API para la gestión de usuarios y publicaciones en GreenForge

Publicaciones

apiPostCommentPost

Comentar en una publicación

Permite a un usuario añadir un comentario en una publicación.

POST

```
/api/post/comment
```

Usage and SDK Samples

- Curl
- Java
- Android
- Obj-C
- JavaScript
- C#
- PHP
- Perl
- Python

```
curl -X POST\
-H "Accept: application/json"\
-H "Content-Type: application/json"\
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/post/comment"
```

Parameters

Body parameters

Name	Description
------	-------------

body *	<div> <div>▼ { }</div> <div> <div>id: ▼ { }</div> <div>integer (int64)</div> <div>ID único del comentario</div> </div> </div> <div> <div>rawPostId: ▼ { }</div> <div>integer (int64)</div> <div>ID del post al que pertenece el comentario</div> </div> <div> <div>rawUserId: ▼ { }</div> <div>integer (int64)</div> <div>ID del usuario que realizó el comentario</div> </div> <div> <div>user: ▼ { }</div> <div> <div>id: ▶ { }</div> <div>integer</div> </div> <div> <div>username: ▶ { }</div> <div>string</div> </div> <div> <div>fullname: ▶ { }</div> <div>string</div> </div> <div> <div>profilephotouri: ▶ { }</div> <div>string</div> </div> </div> <div> <div>tags: ▼ [{ }</div> <div>Lista de etiquetas de usuario en el comentario</div> <div> <div>▼ { }</div> <div> <div>id: ▼ { }</div> <div>integer (int64)</div> <div>ID único de la etiqueta de usuario</div> </div> <div> <div>userTagged: ▼ { }</div> <div> <div>id: ▼ { }</div> <div>integer (int64)</div> <div>ID único del usuario</div> </div> <div> <div>username: ▼ { }</div> <div>string</div> <div>Nombre de usuario único</div> </div> <div> <div>fullname: ▼ { }</div> <div>string</div> <div>Nombre completo del usuario</div> </div> <div> <div>profilephotouri: ▼ { }</div> <div>string (uri)</div> <div>URL de la foto de perfil del usuario</div> </div> </div> </div> </div> <div> <div>content: ▼ { }</div> <div>string</div> <div>Contenido del comentario</div> </div>
--------	---

Responses

Status: 201 - Comentario agregado correctamente

Schema	<div> <div>▼ { }</div> <div> <div>id: ▼ { }</div> <div>integer (int64)</div> <div>ID único del comentario</div> </div> </div> <div> <div>rawPostId: ▼ { }</div> <div>integer (int64)</div> <div>ID del post al que pertenece el comentario</div> </div> <div> <div>rawUserId: ▼ { }</div> <div>integer (int64)</div> <div>ID del usuario que realizó el comentario</div> </div> <div> <div>user: ▼ { }</div> <div> <div>id: ▼ { }</div> <div>integer</div> </div> <div> <div>username: ▼ { }</div> <div>string</div> </div> <div> <div>fullname: ▼ { }</div> <div>string</div> </div> <div> <div>profilephotouri: ▼ { }</div> <div>string</div> </div> </div>
--------	---

```

    id:      ▼ [] integer (int64)
               ID único del usuario

    username: ▼ [] string
               Nombre de usuario único

    fullname: ▼ [] string
               Nombre completo del usuario

    profilephotouri: ▼ [] string (uri)
               URL de la foto de perfil del usuario

  }

  tags: ▼ [ []
           Lista de etiquetas de usuario en el comentario

           ▼ { []
              id:      ▼ [] integer (int64)
                           ID único de la etiqueta de usuario

              userTagged: ▼ { []
                             id:      ▼ [] integer (int64)
                                 ID único del usuario

                             username: ▼ [] string
                                 Nombre de usuario único

                             fullname: ▼ [] string
                                 Nombre completo del usuario

                             profilephotouri: ▼ [] string (uri)
                                 URL de la foto de perfil del usuario

                             }
              }
           ]

  content: ▼ [] string
            Contenido del comentario

}
```

apiPostIdDelete

Eliminar una publicación

Permite a un usuario eliminar una publicación específica por su ID.

DELETE

/api/post/{id}

Usage and SDK Samples

- Curl
- Java
- Android
- Obj-C
- JavaScript
- C#
- PHP
- Perl
- Python

```
curl -X DELETE\  
-H "Accept: application/json"\  
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/post/{id}"
```

Parameters

Path parameters

Name	Description
id*	Long (int64) <i>ID de la publicación a eliminar</i> Required

Responses

Status: 200 - Publicación eliminada correctamente

Schema
boolean

apiPostInteractionsPost

Verificar interacciones del usuario en una publicación

Verifica si un usuario ha dado like o ha comentado en una publicación específica.

POST

```
/api/post/interactions
```

Usage and SDK Samples

Curl	Java	Android	Obj-C	JavaScript	C#	PHP	Perl	Python
------	------	---------	-------	------------	----	-----	------	--------

```
curl -X POST\  
-H "Accept: application/json"\  
-H "Content-Type: application/json"\  
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/post/interactions"
```

Parameters

Body parameters

Name	Description
body *	<div>▼ { []</div> <div>postId: ▼ [] integer (int64) <i>ID de la publicación a verificar</i></div> <div>username: ▼ [] string <i>Nombre de usuario a verificar</i></div> <div>}</div>

Responses

Status: 200 - Estado de interacciones obtenido correctamente

Schema

▼ { }

liked: boolean

commented: boolean

}

apiPostLikePost

Dar o quitar like a una publicación

Permite que un usuario dé o retire un like de una publicación.

POST

/api/post/like

Usage and SDK Samples

CurlJavaAndroidObj-CJavaScriptC#PHPPerlPython

curl -X POST\
-H "Accept: application/json"
-H "Content-Type: application/json"
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/post/like"

Parameters

Body parameters

Name	Description
body *	<div><div>▼ { }</div><div>id: ▼ integer (int64) ID único del like</div><div>rawPostId: ▼ integer (int64) ID del post al que se le ha dado like</div><div>rawUserId: ▼ integer (int64) ID del usuario que ha dado like</div><div>user: ▼ { }</div><div>id: integer</div><div>username: string</div><div>fullname: string</div><div>profilephotouri: string</div><div>}</div></div>

Responses

Status: 200 - Like actualizado correctamente

Schema
boolean

apiPostPost

Crear una nueva publicación

Permite a un usuario crear una nueva publicación.

POST

/api/post

Usage and SDK Samples

Curl	Java	Android	Obj-C	JavaScript	C#	PHP	Perl	Python
------	------	---------	-------	------------	----	-----	------	--------

```
curl -X POST\  
-H "Accept: application/json"\  
-H "Content-Type: application/json"\  
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/post"
```

Parameters

Body parameters

Name	Description
------	-------------

body *

```
▼ { []
  id: ▼ [] integer (int64)
      ID único de la publicación

  user: ▼ { []
          id: ▼ [] integer
          username: ▼ [] string
          fullname: ▼ [] string
          profilephotouri: ▼ [] string
        }

  content: ▼ [] string
           Contenido de la publicación

  imageAttached: ▼ [] string (uri)
                 URL de la imagen adjunta a la publicación

  uploadDate: ▼ [] string (date)
              Fecha de subida de la publicación (YYYY-MM-DD)

  modified: ▼ [] boolean
            Indica si la publicación ha sido modificada

  hashtags: ▼ [ []
             Lista de hashtags asociados a la publicación
             ▼ { []
               id: ▼ [] integer (int64)
                   ID único del hashtag

               name: ▼ [] string
                     Nombre del hashtag
             }
           ]

  rawHashtags: ▼ [ []
               Lista de hashtags en formato de texto
               string
             ]

  likes: ▼ [ []
         Lista de reacciones (likes) en la publicación
         ▼ { []
           id: ▼ [] integer (int64)
               ID único del like

           rawPostId: ▼ [] integer (int64)
                     ID del post al que se le ha dado like

           rawUserId: ▼ [] integer (int64)
                     ID del usuario que ha dado like

           user: ▼ { []
                  id: ▼ [] integer (int64)
                      ID único del usuario

                  username: ▼ [] string
                          Nombre de usuario único

                  fullname: ▼ [] string
                          Nombre completo del usuario

                  profilephotouri: ▼ [] string (uri)
```

```

    }
    }
  ]
  comments: [
    {
      id: integer (int64)
        ID único del comentario

      rawPostId: integer (int64)
        ID del post al que pertenece el comentario

      rawUserId: integer (int64)
        ID del usuario que realizó el comentario

      user: {
        id: integer (int64)
          ID único del usuario

        username: string
          Nombre de usuario único

        fullname: string
          Nombre completo del usuario

        profilephotouri: string (uri)
          URL de la foto de perfil del usuario

      }

      tags: [
        {
          id: integer (int64)
            ID único de la etiqueta de usuario

          userTagged: {
            id: integer (int64)
              ID único del usuario

            username: string
              Nombre de usuario único

            fullname: string
              Nombre completo del usuario

            profilephotouri: string (uri)
              URL de la foto de perfil del usuario

          }

        }

      ]

      content: string
        Contenido del comentario

    }
  ]
}

```

Responses

Status: 201 - Publicación creada correctamente


```
▼ { []
  id: ▼ [] integer (int64)
    ID único de la publicación

  user: ▼ { []
    id: ▼ [] integer (int64)
      ID único del usuario

    username: ▼ [] string
      Nombre de usuario único

    fullname: ▼ [] string
      Nombre completo del usuario

    profilephotouri: ▼ [] string (uri)
      URL de la foto de perfil del usuario

  }

  content: ▼ [] string
    Contenido de la publicación

  imageAttached: ▼ [] string (uri)
    URL de la imagen adjunta a la publicación

  uploadDate: ▼ [] string (date)
    Fecha de subida de la publicación (YYYY-MM-DD)

  modified: ▼ [] boolean
    Indica si la publicación ha sido modificada

  hashtags: ▼ [ []
    Lista de hashtags asociados a la publicación

    ▼ { []
      id: ▼ [] integer (int64)
        ID único del hashtag

      name: ▼ [] string
        Nombre del hashtag

    }

  ]

  rawHashtags: ▼ [ []
    Lista de hashtags en formato de texto

    string

  ]

  likes: ▼ [ []
    Lista de reacciones (likes) en la publicación

    ▼ { []
      id: ▼ [] integer (int64)
        ID único del like

      rawPostId: ▼ [] integer (int64)
        ID del post al que se le ha dado like

      rawUserId: ▼ [] integer (int64)
        ID del usuario que ha dado like

      user: ▼ { []
```

```

    id:      ▼ integer (int64)
              ID único del usuario

    username: ▼ string
              Nombre de usuario único

    fullname: ▼ string
              Nombre completo del usuario

    profilephotouri: ▼ string (uri)
                      URL de la foto de perfil del usuario
  }
}

]

comments: ▼ [ ]
           Lista de comentarios en la publicación
           ▼ { }
             id:      ▼ integer (int64)
                       ID único del comentario

             rawPostId: ▼ integer (int64)
                       ID del post al que pertenece el comentario

             rawUserId: ▼ integer (int64)
                       ID del usuario que realizó el comentario

             user:     ▼ { }
                       id:      ▼ integer (int64)
                                 ID único del usuario

                       username: ▼ string
                                 Nombre de usuario único

                       fullname: ▼ string
                                 Nombre completo del usuario

                       profilephotouri: ▼ string (uri)
                                         URL de la foto de perfil del usuario
             }

           tags: ▼ [ ]
                Lista de etiquetas de usuario en el comentario
                ▼ { }
                  id:      ▼ integer (int64)
                            ID único de la etiqueta de usuario

                  userTagged: ▼ { }
                              id:      ▼ integer (int64)
                                        ID único del usuario

                              username: ▼ string
                                        Nombre de usuario único

                              fullname: ▼ string
                                        Nombre completo del usuario

                              profilephotouri: ▼ string (uri)
                                                URL de la foto de perfil del usuario
                  }
           }
}

]

```

```
        content: {
          type: "string",
          description: "Contenido del comentario"
        }
      }
    ]
  }
```

apiPostPut

Actualizar una publicación

Permite actualizar el contenido de una publicación existente.

PUT

/api/post

Usage and SDK Samples

- Curl
- Java
- Android
- Obj-C
- JavaScript
- C#
- PHP
- Perl
- Python

```
curl -X PUT \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/post"
```

Parameters

Body parameters

Name	Description
------	-------------

body *

```

▼ { []
  id: ▼ [] integer (int64)
    ID único de la publicación

  user: ▼ { []
    id: ▶ [] integer
    username: ▶ [] string
    fullname: ▶ [] string
    profilephotouri: ▶ [] string
  }

  content: ▼ [] string
    Contenido de la publicación

  imageAttached: ▼ [] string (uri)
    URL de la imagen adjunta a la publicación

  uploadDate: ▼ [] string (date)
    Fecha de subida de la publicación (YYYY-MM-DD)

  modified: ▼ [] boolean
    Indica si la publicación ha sido modificada

  hashtags: ▼ [ []
    Lista de hashtags asociados a la publicación
    ▼ { []
      id: ▼ [] integer (int64)
        ID único del hashtag
      name: ▼ [] string
        Nombre del hashtag
    }
  ]

  rawHashtags: ▼ [ []
    Lista de hashtags en formato de texto
    string
  ]

  likes: ▼ [ []
    Lista de reacciones (likes) en la publicación
    ▼ { []
      id: ▼ [] integer (int64)
        ID único del like
      rawPostId: ▼ [] integer (int64)
        ID del post al que se le ha dado like
      rawUserId: ▼ [] integer (int64)
        ID del usuario que ha dado like
      user: ▼ { []
        id: ▼ [] integer (int64)
          ID único del usuario
        username: ▼ [] string
          Nombre de usuario único
        fullname: ▼ [] string
          Nombre completo del usuario
        profilephotouri: ▼ [] string (uri)

```

```

    }
    }
  ]
  comments: [
    {
      id: integer (int64)
        ID único del comentario

      rawPostId: integer (int64)
        ID del post al que pertenece el comentario

      rawUserId: integer (int64)
        ID del usuario que realizó el comentario

      user: {
        id: integer (int64)
          ID único del usuario

        username: string
          Nombre de usuario único

        fullname: string
          Nombre completo del usuario

        profilephotouri: string (uri)
          URL de la foto de perfil del usuario

      }

      tags: [
        {
          id: integer (int64)
            ID único de la etiqueta de usuario

          userTagged: {
            id: integer (int64)
              ID único del usuario

            username: string
              Nombre de usuario único

            fullname: string
              Nombre completo del usuario

            profilephotouri: string (uri)
              URL de la foto de perfil del usuario

          }

        }

      ]

      content: string
        Contenido del comentario

    }
  ]
}

```

Responses

Status: 200 - Publicación actualizada correctamente

```
▼ { []
  id: ▼ [] integer (int64)
    ID único de la publicación

  user: ▼ { []
    id: ▼ [] integer (int64)
      ID único del usuario

    username: ▼ [] string
      Nombre de usuario único

    fullname: ▼ [] string
      Nombre completo del usuario

    profilephotouri: ▼ [] string (uri)
      URL de la foto de perfil del usuario

  }

  content: ▼ [] string
    Contenido de la publicación

  imageAttached: ▼ [] string (uri)
    URL de la imagen adjunta a la publicación

  uploadDate: ▼ [] string (date)
    Fecha de subida de la publicación (YYYY-MM-DD)

  modified: ▼ [] boolean
    Indica si la publicación ha sido modificada

  hashtags: ▼ [ []
    Lista de hashtags asociados a la publicación

    ▼ { []
      id: ▼ [] integer (int64)
        ID único del hashtag

      name: ▼ [] string
        Nombre del hashtag

    }

  ]

  rawHashtags: ▼ [ []
    Lista de hashtags en formato de texto

    string

  ]

  likes: ▼ [ []
    Lista de reacciones (likes) en la publicación

    ▼ { []
      id: ▼ [] integer (int64)
        ID único del like

      rawPostId: ▼ [] integer (int64)
        ID del post al que se le ha dado like

      rawUserId: ▼ [] integer (int64)
        ID del usuario que ha dado like

      user: ▼ { []
```

```

        id:          ▼ [] integer (int64)
                       ID único del usuario

        username:    ▼ [] string
                       Nombre de usuario único

        fullname:    ▼ [] string
                       Nombre completo del usuario

        profilephotouri: ▼ [] string (uri)
                       URL de la foto de perfil del usuario
    }

}

]

comments: ▼ [ []
            Lista de comentarios en la publicación

            ▼ { []
                id:          ▼ [] integer (int64)
                               ID único del comentario

                rawPostId:  ▼ [] integer (int64)
                               ID del post al que pertenece el comentario

                rawUserId:  ▼ [] integer (int64)
                               ID del usuario que realizó el comentario

                user:        ▼ { []
                    id:          ▼ [] integer (int64)
                                   ID único del usuario

                    username:    ▼ [] string
                                   Nombre de usuario único

                    fullname:    ▼ [] string
                                   Nombre completo del usuario

                    profilephotouri: ▼ [] string (uri)
                                   URL de la foto de perfil del usuario
                }

            tags: ▼ [ []
                    Lista de etiquetas de usuario en el comentario

                    ▼ { []
                        id:          ▼ [] integer (int64)
                                       ID único de la etiqueta de usuario

                        userTagged: ▼ { []
                            id:          ▼ [] integer (int64)
                                           ID único del usuario

                            username:    ▼ [] string
                                           Nombre de usuario único

                            fullname:    ▼ [] string
                                           Nombre completo del usuario

                            profilephotouri: ▼ [] string (uri)
                                           URL de la foto de perfil del usuario
                        }
                    }
            ]
        }
    ]

```

```
        content: ▼ [] string
                  Contenido del comentario
      }
    ]
  }
}
```

apiPostUsernameGet

Obtener publicaciones de un usuario

Retorna una lista de publicaciones realizadas por un usuario específico.

GET

```
/api/post/{username}
```

Usage and SDK Samples

CurlJavaAndroidObj-CJavaScriptC#PHPPerlPython

```
curl -X GET\
-H "Accept: application/json"\
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/post/{username}"
```

Parameters

Path parameters

Name	Description
username*	String Nombre de usuario del creador de las publicaciones Required

Responses

Status: 200 - Lista de publicaciones del usuario obtenida correctamente

Schema

```
▼ [ []
  ▼ { []
    id: ▼ [] integer (int64)
         ID único de la publicación

    user: ▼ { []
            id: ▼ [] integer (int64)
                 ID único del usuario

            username: ▼ [] string
                       Nombre de usuario único

            fullname: ▼ [] string
                      Nombre completo del usuario
```



```

    profilephotouri: ▼ [] string (uri)
                        URL de la foto de perfil del usuario
  }
content: ▼ [] string
          Contenido de la publicación

imageAttached: ▼ [] string (uri)
                URL de la imagen adjunta a la publicación

uploadDate: ▼ [] string (date)
             Fecha de subida de la publicación (YYYY-MM-DD)

modified: ▼ [] boolean
          Indica si la publicación ha sido modificada

hashtags: ▼ [ []
            Lista de hashtags asociados a la publicación
            ▼ { []
              id: ▼ [] integer (int64)
                  ID único del hashtag

              name: ▼ [] string
                   Nombre del hashtag
            }
          ]
rawHashtags: ▼ [ []
              Lista de hashtags en formato de texto
              string
            ]
likes: ▼ [ []
        Lista de reacciones (likes) en la publicación
        ▼ { []
          id: ▼ [] integer (int64)
              ID único del like

          rawPostId: ▼ [] integer (int64)
                    ID del post al que se le ha dado like

          rawUserId: ▼ [] integer (int64)
                    ID del usuario que ha dado like

          user: ▼ { []
                id: ▼ [] integer (int64)
                    ID único del usuario

                username: ▼ [] string
                        Nombre de usuario único

                fullname: ▼ [] string
                        Nombre completo del usuario

                profilephotouri: ▼ [] string (uri)
                                URL de la foto de perfil del usuario
              }
        }
      ]
comments: ▼ [ []
           Lista de comentarios en la publicación

```

```

▼ { []
  id: ▼ [] integer (int64)
      ID único del comentario

  rawPostId: ▼ [] integer (int64)
              ID del post al que pertenece el comentario

  rawUserId: ▼ [] integer (int64)
              ID del usuario que realizó el comentario

  user: ▼ { []
          id: ▼ [] integer (int64)
              ID único del usuario

          username: ▼ [] string
                    Nombre de usuario único

          fullname: ▼ [] string
                    Nombre completo del usuario

          profilephotouri: ▼ [] string (uri)
                           URL de la foto de perfil del usuario
        }

  tags: ▼ [ []
          Lista de etiquetas de usuario en el comentario

          ▼ { []
            id: ▼ [] integer (int64)
                ID único de la etiqueta de usuario

            userTagged: ▼ { []
                        id: ▼ [] integer (int64)
                            ID único del usuario

                        username: ▼ [] string
                                Nombre de usuario único

                        fullname: ▼ [] string
                                Nombre completo del usuario

                        profilephotouri: ▼ [] string (uri)
                                         URL de la foto de perfil del usuario
                      }
          }
        ]

  content: ▼ [] string
            Contenido del comentario
}
]
}
]

```

Usuarios

apiUserFollowPost

Seguir o dejar de seguir a un usuario

Permite que un usuario siga o deje de seguir a otro usuario.

POST

```
/api/user/follow
```

Usage and SDK Samples

- Curl
- Java
- Android
- Obj-C
- JavaScript
- C#
- PHP
- Perl
- Python

```
curl -X POST\  
-H "Accept: application/json"\  
-H "Content-Type: application/json"\  
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/user/follow"
```

Parameters

Body parameters

Name	Description
body *	boolean

Responses

Status: 200 - Acción completada correctamente

Schema

boolean

Status: 400 - Datos inválidos

apiUserPut

Actualizar la información de un usuario

Permite actualizar la información básica de un usuario.

PUT

```
/api/user
```

Usage and SDK Samples

```
curl -X PUT \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/user"
```

Parameters

Body parameters

Name	Description
body *	<div><div>▼ { }</div><div><div>id: ▼ integer (int64) ID único del usuario</div><div>username: ▼ string Nombre de usuario único</div><div>fullname: ▼ string Nombre completo del usuario</div><div>profilephotouri: ▼ string (uri) URL de la foto de perfil del usuario</div></div><div>}</div></div>

Responses

Status: 200 - Usuario actualizado correctamente

Schema
<div><div>▼ { }</div><div><div>id: ▼ integer (int64) ID único del usuario</div><div>username: ▼ string Nombre de usuario único</div><div>fullname: ▼ string Nombre completo del usuario</div><div>profilephotouri: ▼ string (uri) URL de la foto de perfil del usuario</div></div><div>}</div></div>

Status: 400 - Datos inválidos

apiUserUsernameFollowedGet

Obtener seguidos de un usuario

Retorna la lista de usuarios que sigue un usuario específico.

GET

/api/user/{username}/followed

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X GET\
-H "Accept: application/json"\
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/user/{username}/followed"
```

Parameters

Path parameters

Name	Description
username*	String <i>Nombre de usuario del usuario a consultar</i> Required

Responses

Status: 200 - Lista de seguidos obtenida correctamente

Schema

```
▼ [ {}
  ▼ { {}
    id:      ▼ {} integer (int64)
              ID único del usuario

    username: ▼ {} string
               Nombre de usuario único

    fullname: ▼ {} string
               Nombre completo del usuario

    profilephotouri: ▼ {} string (uri)
                     URL de la foto de perfil del usuario

  }
]
```

Status: 404 - Usuario no encontrado

apiUserUsernameFollowersGet

Obtener seguidores de un usuario

Retorna la lista de seguidores de un usuario específico.

GET

/api/user/{username}/followers

Usage and SDK Samples

- Curl
- Java
- Android
- Obj-C
- JavaScript
- C#
- PHP
- Perl
- Python

```
curl -X GET\  
-H "Accept: application/json"\  
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/user/{username}/followers"
```

Parameters

Path parameters

Name	Description
username*	String Nombre de usuario del usuario a consultar Required

Responses

Status: 200 - Lista de seguidores obtenida correctamente

Schema

▼ [{}

▼ { {}

id: ▼ {} integer (int64)
ID único del usuario

username: ▼ {} string
Nombre de usuario único

fullname: ▼ {} string
Nombre completo del usuario

profilephotouri: ▼ {} string (uri)
URL de la foto de perfil del usuario

}

]

Status: 404 - Usuario no encontrado

apiUserUsernameGet

Obtener un usuario por nombre de usuario

Retorna los detalles del usuario basado en su nombre de usuario.

GET

```
/api/user/{username}
```

Usage and SDK Samples

- Curl
- Java
- Android
- Obj-C
- JavaScript
- C#
- PHP
- Perl
- Python

```
curl -X GET\
-H "Accept: application/json"\
"https://virtserver.swaggerhub.com/KevinGonzalez/GreenForge/1.0.0/api/user/{username}"
```

Parameters

Path parameters

Name	Description
username*	String Nombre de usuario del usuario a buscar Required

Responses

Status: 200 - Usuario encontrado

Schema

▼ { }

id: ▼ integer (int64)

username: string

email: ▼ string (email)

fullname: string

birthdate: ▼ string (date)

phone: ▼ integer (int64)

profilephotouri: ▼ string (uri)

biography: string

followers: ▼ [{ }

followed: ▼ [{ }

id: ▼ integer (int64)
ID único del usuario

username: ▼ string
Nombre de usuario único

fullname: ▼ string
Nombre completo del usuario

profilephotouri: ▼ string (uri)
URL de la foto de perfil del usuario

id: ▼ integer (int64)
ID único del usuario

username: ▼ string
Nombre de usuario único

fullname: ▼ string
Nombre completo del usuario

profilephotouri: ▼ string (uri)
URL de la foto de perfil del usuario

```
} ]
```

Status: 404 - Usuario no encontrado

Suggestions, contact, support and error reporting;

Information URL: <https://helloverb.com> (<https://helloverb.com>)

Contact Info: hello@helloverb.com (hello@helloverb.com)

All rights reserved

<http://apache.org/licenses/LICENSE-2.0.html>