API description template

*Please see the template tables (for each verb) that you can use to create the API specification that fits the given API (posted on Blackboard). To see an example for each Verb, see the result for the Server-side homework week 1.*

*Instead of this document and templates, you can use other tools to create the API specification as well. An example of such a tool is the Swagger Editor (*[*https://editor.swagger.io/*](https://editor.swagger.io/)*).*

Table of Contents

[1. Class diagram 3](#_Toc71106369)

[2. GET requests 4](#_Toc71106370)

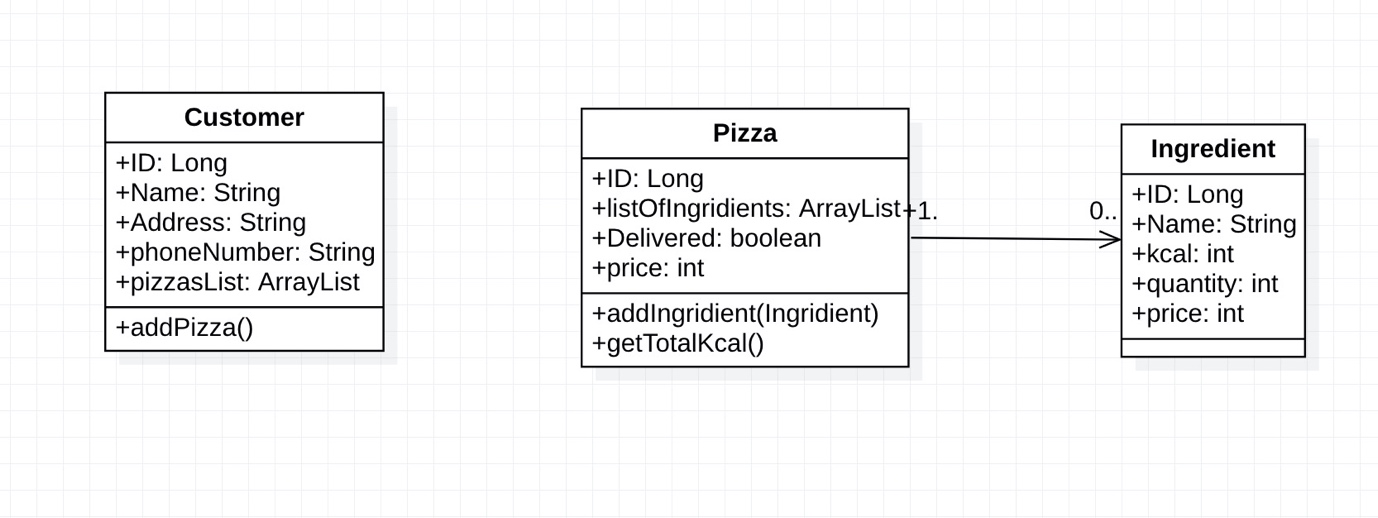
[3. POST requests 5](#_Toc71106371)

[4. PUT requests 6](#_Toc71106372)

[5. DELETE requests 7](#_Toc71106373)

# Class diagram

*Put your class diagram here. Also write some text that describes what is going on in your class diagram. For example, point out what the different unique identifiers are for each class*

*and how the classes are related to each other.*

# GET requests

*Add your requests here. Copy-paste the template for each different request.*

Template GET table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/customers** | | | |
| Get a list of all customers | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | List of customers can be empty | |
|  |  | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/customers/{id}** | | | |
| Get one customer, based on id | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | Id of the customer to find |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | The found customer based on ID | |
| 404 | Thrown if no customer was found | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/customers/{id}/pizzas** | | | |
| Get a list of pizzas for a specific customer, based on id | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | Id of the customer to find |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | List of pizzas for a specific customer | |
| 404 | Thrown if no customer was found | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/pizzas** | | | |
| Get a list of all pizzas, independent of customer | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | Delivered | query | Filter on delivered state, either true or false |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | List of pizzas, independent on customers. Can be empty | |
|  |  | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/pizzas/{id}** | | | |
| Get one pizza, based on id | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | Id of the pizza to find |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | The found pizza based on ID | |
| 404 | Thrown if no pizza was found | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/ pizzas/{id}/ ingredients** | | | |
| Get a list of ingredients for a specific pizza, based on id | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | Id of the pizza to find |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | List of ingredients for a specific pizza | |
| 404 | Thrown if no pizza was found | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GET** | **/ pizzas/{id}/ingredients/{id}** | | | |
| Get a list of ingredients for a specific pizza, based on id | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | Id of the pizza to find |
| id | path | Id of the ingredient to find |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | List of ingredients for a specific pizza | |
| 404 | Thrown if no pizza was found | |
|  |  | |
|  |  | |
|  |  | |

# POST requests

*Add your requests here. Copy-paste the template for each different request.*

Template POST table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **POST** | **/customers** | | | |
| Create new customer | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | Customer | body | The customer that needs to be added |
|  |  |  |
|  |  |  |
|  |  |  |
|  | (path / body) |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 201 | Customer created successfully | |
| 400 | Thrown when something was wrong with customer body | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **POST** | **/customers/{id}/pizzas** | | | |
| Create new pizza and assign it to the customer | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
|  | | id | path | Id of customer |
| pizza | Body | The pizza |
|
|
|
|
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 201 | The pizza was created successfully | |
| 400 | Thrown when something was wrong with pizza body | |
| 404 | Customer id not found | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **POST** | **/pizzas** | | | |
| Create new pizza | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | Pizza | body | The pizza that needs to be added |
|  |  |  |
|  |  |  |
|  |  |  |
|  | (path / body) |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 201 | Pizza created successfully | |
| 400 | Thrown when something was wrong with pizza’s body | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **POST** | **/pizzas/{id}/ingredients** | | | |
| Create new ingredient and assign it to the pizza | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
|  | | id | path | Id of pizza |
| ingredient | Body | The ingredient |
|
|
|
|
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 201 | The ingredient was created successfully | |
| 400 | Thrown when something was wrong with ingredient body | |
| 404 | Pizza id not found | |
|  |  | |
|  |  | |

# PUT requests

*Add your requests here. Copy-paste the template for each different request.*

Template PUT table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PUT** | **/customers/{id}** | | | |
| Update a customer based on id | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | ID of the customer that needs to be updated |
| customer | body | Updated customer |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Updated customer is retuned if successful | |
| 400 | Thrown if customer body was wrong or customer id in body did not matched customer id in path | |
| 404 | Thrown if customer was not found | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PUT** | **/pizzas/{id}** | | | |
| Update a pizza based on id | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | ID of the pizza that needs to be updated |
| pizza | body | Updated pizza |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Updated pizza is retuned if successful | |
| 400 | Thrown if pizza body was wrong or pizza id in body did not matched pizza id in path | |
| 404 | Thrown if pizza was not found | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PUT** | **/pizzas/{id}/ingredients/{id}** | | | |
| Update an ingredient based on id | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | ID of the ingredient that needs to be updated |
| ingredient | body | Updated ingredient |
| id | path | Id of the pizza with that ingredient |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 200 | Updated ingredient is retuned if successful | |
| 400 | Thrown if ingredient body was wrong or ingredient id in body did not matched ingredient id in path | |
| 404 | Thrown if ingredient was not found | |
|  |  | |
|  |  | |

# DELETE requests

*Add your requests here. Copy-paste the template for each different request.*

Template PUT table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DELETE** | **/customers/{id}** | | | |
| Delete a customer (and all his pizzas) | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | ID of customer to be deleted |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 204 | Returned when deleting of customer was successful | |
| 404 | Thrown if no customer was found with that ID | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DELETE** | **/pizzas/{id}** | | | |
| Delete a pizza (and all its ingredients) | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | ID of pizza to be deleted |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 204 | Returned when deleting of pizza was successful | |
| 404 | Thrown if no pizza was found with that ID | |
|  |  | |
|  |  | |
|  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DELETE** | **/pizzas/{id}/ingredients/{id}** | | | |
| Delete an ingredient | | | | |
|  | | | | |
| **Parameters:** | | **Name** | **Type** | **Description** |
| *Add a \* to the name of required parameters.* | | id | path | ID of ingredient to be deleted |
| id | path | ID of pizza with needed ingredient to be deleted |
|  |  |  |
|  |  |  |
|  |  |  |
| **Responses:** | | **Code** | **Description / example if successful** | |
|  | | 204 | Returned when deleting of ingredient was successful | |
| 404 | Thrown if no ingredient was found with that ID | |
|  |  | |
|  |  | |
|  |  | |