

Smart Parking

Generated by Doxygen 1.9.1

1 Smart Parking documentation	1
1.1 Introduction	1
1.2 Installation	1
1.2.1 To run the project	1
1.2.2 To compile the project	1
2 File Index	3
2.1 File List	3
3 File Documentation	5
3.1 project.c File Reference	5
3.1.1 Detailed Description	6
3.1.2 Function Documentation	6
3.1.2.1 all_slots()	6
3.1.2.2 all_slots_graphic()	6
3.1.2.3 change_slot()	6
3.1.2.4 check_slot()	7
3.1.2.5 create_parking()	7
3.1.2.6 main()	7
3.1.2.7 numberOfAvailable()	8
Index	9

Chapter 1

Smart Parking documentation

1.1 Introduction

This project was created for the NF05A course

Original instructions : [Project_Smart_Parking.pdf](#)

1.2 Installation

1.2.1 To run the project

the binaries are available for Windows (project.exe) and for Linux (project.o)

1.2.2 To compile the project

you just have to compile [project.c](#) (with gcc, preferably)

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

project.c	This is the main file of the project	5
---------------------------	--	---

Chapter 3

File Documentation

3.1 project.c File Reference

This is the main file of the project.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
```

Functions

- void `create_parking` ()
Prodedure that creates the file of the parking
- int `check_slot` (int slot_number)
This function checks if a specific slot is taken or not.
- void `all_slots` ()
This procedure displays every slot that is available.
- void `all_slots_graphic` ()
*This procedure displays every slot in a graphic way
the "graphic" table is created thanks to the extended ascii character, which are supported by the c language and the printf function.*
- void `numberOfAvailable` ()
This procedure displays the number of available slots.
- void `change_slot` (int slot, char state)
The goal of this procedure is to change the value of a specific slot.
- void `makeReservation` ()
This procedure allows the user to book a specific slot if it is not already taken.
- void `free_slot` ()
This procedure allows the user to free a specific slot if it is not already available.
- int `main` ()
this is where the program starts

3.1.1 Detailed Description

This is the main file of the project.

Author

THARAUD Valentin & SAUVAGE Eli

Version

1.0

Date

25/04/2021

3.1.2 Function Documentation

3.1.2.1 all_slots()

```
void all_slots ( )
```

This procedure displays every slot that is available.

Note

the file has to exist for this function to work

3.1.2.2 all_slots_graphic()

```
void all_slots_graphic ( )
```

This procedure displays every slot in a graphic way
the "graphic" table is created thanks to the extended ascii character, which are supported by the c language and the printf function.

Note

the file has to exist for this function to work

3.1.2.3 change_slot()

```
void change_slot (
    int slot,
    char state )
```

The goal of this procedure is to change the value of a specific slot.

Parameters

<i>slot</i>	this is the slot to change
<i>state</i>	this is the new state. Its value should be either '0' or '1'

Note

the file has to exist for this function to work

3.1.2.4 check_slot()

```
int check_slot (
    int slot_number )
```

This function checks if a specific slot is taken or not.

Parameters

<i>slot_number</i>	the number of the slot for which the availability is checked
--------------------	--

Returns

0 if available, 1 if taken

Note

the file has to exist for this function to work

3.1.2.5 create_parking()

```
void create_parking ( )
```

Prodedure that creates the file of the parking

by default, the file is filled with 0, which mean that every slot is available

3.1.2.6 main()

```
int main ( )
```

this is where the program starts

Returns

0 for "no error" code p a list of every possible option is printed to the user

3.1.2.7 numberOfAvailable()

```
void numberOfAvailable ( )
```

This procedure displays the number of available slots.

Note

the file has to exist for this function to work

Index

- all_slots
 - project.c, [6](#)
- all_slots_graphic
 - project.c, [6](#)
- change_slot
 - project.c, [6](#)
- check_slot
 - project.c, [7](#)
- create_parking
 - project.c, [7](#)
- main
 - project.c, [7](#)
- numberOfAvailable
 - project.c, [7](#)
- project.c, [5](#)
 - all_slots, [6](#)
 - all_slots_graphic, [6](#)
 - change_slot, [6](#)
 - check_slot, [7](#)
 - create_parking, [7](#)
 - main, [7](#)
 - numberOfAvailable, [7](#)