dsfp Documentation

Release 0.0.1

You'd better not know

March 24, 2014

Contents

Contents:

CHAPTER 1

Datasheet

`Base File Index` equals to *Start block* **0x2c0** (**BFI**)

1.1 Headers format and structures

Note: Information about **data type** you can access with official python documentation

Offset (from the begging)	Size	Data Type	Explain	Value
			File General Header	
0x00	4 bytes	c_char_p	Probably file format container	
0x04	4 bytes	c_uint32	could be a separator	**x00x00**
0x08	4 bytes	c_uint32	Unknown data	
0x0c	4 bytes	c_uint32	could be real block data stored in whole file	hex: **0x**
0x10	4 bytes	c_uint32	Unknown data	
0x14	4 bytes	c_uint32	could be a separator	**x00x00**
0x18	8 bytes	c_char_p	It seems a version of document or application	string:
0x20	4 bytes	c_uint32	Unknown data	hex: **0x**
0x24	4 bytes	c_uint32	could be a separator	**x00x00**
0x28	4 bytes	c_uint32	First data block offset from the of the file *Start block*	hex: **0x**
0x2c	4 bytes	c_uint32	could be a separator	**x00x00**
0x30	4 bytes	c_uint32	Unknown data (0x0120->8193)	
0x34, **0x38**, **0x3c**	4 bytes * 3	c_uint32	Unknown block of data:	**x00x00**
			Start containers header metadata (11 blocks) (*datasheet below represe*	
0x40	4 bytes	c_uint32	Container Header separator High	hex: **0x**
0x44	4 bytes	c_uint32	Container Header separator Low	hex: **0x**
			End of container header metadata	
Containers data 1 block example represented				
0x48	8 bytes	c_ulong	Block data size (in bytes)	
0x50	4 bytes	c_uint32	data block start offset from the file beginning	
0x54	4 bytes	c_uint32	Unknown data, could be an some sort of offsets ?	
0x58	4 bytes	c_uint32	Skip the amount of bytes block (4h->4)	hex: **0x**
0x5c	4 bytes	c_uint32	could be a separator useless block or just 8 byte for **0x58** block	**x00x00**
			End of Containers header data	
			End of Containers	
0x1A0 => 0x2Bf `USER_DATA xxx` representation in following format				
0x1A0	26 bytes	c_char_p	string sequence	USER_
0x1BA	26 bytes	c_char_p	string sequence	USER_

Table 1.1 – continued from previous page

Offset (from the begging)	Size	Data Type	Explain	Value
0x2bc	4 bytes	c_uint32	**And so on until** `USER_DATA 010\x00` **sequence** separator **End of File General Header**	**x00x00**

1.2 Slot block

Slot block data contains very useful information, stats, items, deaths count and so on

Offset (from the begging of slot block)	Size	Data Type	Explain	Value
0x6c	4 Bytes	c_uint32	Health bar current	Depends on stats **600**
0x70	4 Bytes	c_uint32	Health bar	Depends on stats **600**
0x88	4 Bytes	c_uint32	Stamina	Depends on stats max: **160**
0x8c	4 Bytes	c_uint32	Stamina 2	Depends on stats max: **160**
0x90	4 Bytes	c_uint32	Stamina 3	Depends on stats max: **160**
0x98	4 Bytes	c_uint32	Vitality	Depends on level and so on (**30**
0xa0	4 Bytes	c_uint32	Attunement	Depends on level and so on (**10**
0xa8	4 Bytes	c_uint32	Endurance	Depends on level and so on (**50**
0xb0	4 Bytes	c_uint32	Strength	Depends on level and so on (**50**
0xb8	4 Bytes	c_uint32	Dextirity	Depends on level and so on (**20**
0xc0	4 Bytes	c_uint32	Intelligence	Depends on level and so on (**10**
0xc8	4 Bytes	c_uint32	Faith	Depends on level and so on (**10**
0xd8	4 Bytes	c_uint32	Humanity	Depends on level and so on (**00**
0xe0	4 Bytes	c_uint32	Resistance	Depends on level and so on (**11**
0xe8	4 Bytes	c_uint32	Level	Depends on level and so on (**11**
0xec	4 Bytes	c_uint32	Souls	Depends on level and so on (**93**
0xf0	4 Bytes	c_uint32	Earned souls	Depends on level and so on (**99**
0x100	28 Bytes	c_char_p	Character's name	Depends on your imagination (
0x122	1 Bytes	c_bool	Male	Depends on your choice (**1**)
0x126	1 Bytes	c_bool	Class	Depends on your choice (**6**)
0x127	1 Bytes	c_bool	Body	Depends on your choice (**0**)
0x128	1 Bytes	c_bool	Gift	Depends on your choice (**1**)
0x16c	1 Bytes	c_bool	Face	Depends on your choice (**6**)
0x16d	1 Bytes	c_bool	Hairs	Depends on your choice (**6**)
0x16e	1 Bytes	c_bool	Color	Depends on your choice (**6**)
skip				
0x2c0	4 Bytes	c_uint32	First quick slot item type	Any accessible item type (203
0x2c0	4 Bytes	c_uint32	First quick slot item type	Any accessible item type (203
0x2c4	4 Bytes	c_uint32	Second quick slot item type	Any accessible item type (203
0x2c8	4 Bytes	c_uint32	Third quick slot item type	Any accessible item type (203
0x2cc	4 Bytes	c_uint32	Fourth quick slot item type	Any accessible item type (203
0x2d0	4 Bytes	c_uint32	Fifth quick slot item type	Any accessible item type (203
skip				
0x1f128	4 Bytes	c_uint32	Deaths	Depends on your skill :D (**155**

1.3 Start block

Note: **0x2c** starts with first character data. Character data is under heavily process of reverse engineering, so there is only some simple bunch of data represented withing different blocks/structures such like as *Stats*, *Items*

1.4 Stats

Note: All character statistics depending on items/primary stats (vitality->resistance) are dynamicly-calculated type of data. So there's no reason to store it or modify it via runtime. Therefore only two of this type of data present in the table below. it's Capacity and Physical defence.

Note: General stats such like as Vitality, Attunement, Strength and so on stored in RAM with given offsets listed in 3rd table column. But their real address slightly different from save file one, be warned. For example if Vitality has **0x7C563C00** address then Attunement is certainly with 0x08 offset further and stored into **0x7C563C08**

Parameter	Amount of memory entries (ingame)	Amount of store entries (save file)
Stamina	3 slots, 4 byte	BFI + **0x88** (and **0x8c**, **0x90**)
Vitality	1 slot, 4 byte	BFI + **0x98**
Attunement	1 slot, 4 byte	BFI + **0xa0**
Endurance	1 slot, 4 byte	BFI + **0xa8**
Strength	1 slot, 4 byte	BFI + **0xb0**
Dexterity	1 slot, 4 byte	BFI + **0xb8**
Intel.	1 slot, 4 byte	BFI + **0xc0**
Faith	1 slot, 4 byte	BFI + **0xc8**
Unknown	1 slot, 4 byte (unsigned)	BFI + **0xd0**
Humanity	1 slot, 4 byte	BFI + **0xd8**
Resistance	1 slot, 4 byte	BFI + **0xe0**
Level	1 slot, 4 byte	BFI + **0xe8**
Souls	1 slot, 4 byte	BFI + **0xec**
Collected Souls	1 slot, 4 byte	BFI + **0xf0**
Name	1 slot, 28 byte 13*2 = name, 1 = **0x00**	BFI + **0x100**
Gender (male)	1 slot, 1 byte 0 - female, 1 - male	BFI + **0x122**
Class	1 slot, 1 byte	BFI + **0x126**
Body	1 slot, 1 byte	BFI + **0x127**
Gift	1 slot, 1 byte	BFI + **0x128**
Face	1 slot, 1 byte	BFI + **0x16c**
Haris	1 slot, 1 byte	BFI + **0x16d**
Color	1 slot, 1 byte	BFI + **0x16e**
Deaths	1 slot, 1 byte	BFI + **0x1f128**
Dynamicly-calculated data		
Capacity	2 slots 4 byte float (evaluateable)	Offsets don't concure with memory ones
Physical defence	6 slots 4 byte (evaluateable)	Offsets do not concure with memory offsets

1.5 Items

Item	Amount of memory entries (ingame)	Amount of store entries (save file)
Flask of estus	3 slot, 1 active -> 2 depends on active	1 slot: 0x60FF8(random)

1.5.1 Items store structure

```
from ctypes import *
item_proto_structure = {'type': c_uint32, 'amount': c_uint32}
```

Item Type	Ingame Item title
	General items
0x65 (100)	White chalk
0x66 (101)	Red chalk
0x67 (102)	Red eye
0x68 (103)	Black Separation Crystal
0x6a (106)	Orange chalk
0x6c (108)	Book of the Guilty
0x6d (109)	Death eye
0x6f (111)	Cracked red eye orb
0x70 (112)	Servant Roster
0x71 (113)	Blue eye orb
0x72 (114)	Dragon eye orb
0x73 (115)	Black eye orb
0x74 (116)	Black eye orb #2
0x75 (117)	Mark of the darkness
0x76 (118)	Violet coward crystal
0x8 (200)	Estus Flask (empty)
0xC9 (201)	Estus Flask (active)
0xCA (202)	Estus Flask (empty)
0xCB (203)	Estus Flask +1
0xCC (204)	Estus Flask +1 (empty)
0xCD (205)	Estus Flask +2
0xCE (206)	Estus Flask +3 (empty)
0xCF (207)	Estus Flask +3
0xD0 (208)	Estus Flask +4 (empty)
0xD1 (209)	Estus Flask +4
0xD2 (210)	Estus Flask +5 (empty)
0xD3 (211)	Estus Flask +5
0xD4 (212)	Estus Flask +6 (empty)
0xD5 (213)	Estus Flask +6
0xD6 (214)	Estus Flask +7 (empty)
0xD7 (215)	Estus Flask +7
	General Items #2
0xDC (220)	Silver neckle
0xE6 (230)	Elisabeth Mushroom
0xF0 (240)	Heaven's blessing
0x104 (260)	Green flower
0x10e (270)	Red moss
0x10f (271)	Purple moss
0x110 (272)	Purple blooming moss
0x110 (274)	Stone of repel
0x113 (275)	Eggs medicine
0x118 (280)	Fixment powder
0x122 (290)	Throwing knife
0x123 (291)	Poisoned throwing knife
0x124 (292)	Fire bomb
0x125 (293)	Manure pie

Continued on next page

Table 1.3 – continued from previous page

Item Type	Ingame Item title
0x126 (294)	Nice skull
0x10e (296)	LLoyd's charm
0x129 (297)	Black bomb
0x136 (310)	Black gum (resin)
0x137 (311)	Golden gum (resin)
0x138 (312)	Short duration damnation
0x139 (313)	Rotten gum (resin)
0x14a (330)	Bone of return
0x15e (350)	Humanity
0x172 (370)	Shiny stone
0x172 (371)	Field glasses
0x175 (373)	List of charge
0x176 (374)	Mark (sign) of retribution
0x177 (375)	Medal of the light
0x178 (376)	Neckle
0x179 (377)	Stone with gragon head
0x17a (378)	Stone with dragon body
0x17c (380)	Trash
0x17d (381)	Copper coin
0x17e (382)	Silver coin
0x17f (383)	Golden coin
0x180 (384)	Strange doll
0x181 (385)	Dried finger
0x186 (390)	Keeper of fire soul
0x187 (391)	Keeper of fire soul (no mistake there's one more)
0x188 (392)	Keeper of fire soul (no mistake there's one more)
0x189 (393)	Keeper of fire soul (no mistake there's one more)
0x18a (394)	Keeper of fire soul (no mistake there's one more)
0x18b (395)	Keeper of fire soul (no mistake there's one more)
0x18c (396)	Keeper of fire soul (no mistake there's one more)
0x190 (400)	Soul of the lost undead
0x18e (401)	Big soul of the lost undead
0x18f (402)	Soul of nameless soldier
0x190 (403)	Big soul of nameless soldier
0x191 (404)	Soul of a Proud Knight
0x192 (405)	Big soul of a Proud Knight
0x193 (406)	Soul of a Brave Warrior
0x194 (407)	Big soul of a Brave Warrior
0x195 (408)	Soul of a Hero
0x196 (409)	Soul of a Great Hero
0x1f4 (500)	Humanity
0x1f5 (501)	Double humanity
0x1fe (510)	Sign "hello"
0x1ff (511)	Sign "thanks"
0x200 (512)	Sign "perfect" (gj, gd, whatever)
0x201 (513)	Sign "sorry"
0x2BC (700)	Soul of Sif
0x2BD (701)	Soul of Quelaag
0x2BE (702)	Soul of Gwynn
0x2BF (703)	Soul of Iron Golem
0x2C0 (704)	Soul of Ornstein

Continued on next page

Table 1.3 – continued from previous page

Item Type	Ingame Item title
0x2C1 (705)	Soul of Moonlight butterfly
0x2C2 (706)	Soul of Smough
0x2C3 (707)	Soul of Priscilla
0x2C4 (708)	Soul of Gwyndolin
0x2C5 (709)	Soul of Sentinel
0x2C6 (710)	Soul of Artorias
0x2C7 (710)	Soul of Manus
0x320 (800)	Big coal
0x321 (801)	Very big coal
0x322 (802)	Crystal coal
0x326 (806)	Big magic coal
0x327 (807)	Charmed coal
0x328 (808)	Heaven's coal
0x329 (809)	Big heaven's coal
0x32a (810)	Coal of darkness
0x32c (812)	Big coal of fire
	Forge items #2
0x3e8 (1000)	Titanite Shard
0x3f2 (1010)	Large Titanite Shard
0x3fc (1020)	Green Titanite Shard
0x406 (1030)	Titanite chunk
0x410 (1040)	Blue Titanite chunk
0x41a (1050)	White Titanite chunk
0x424 (1060)	Red Titanite chunk
0x42e (1070)	Titanite slab
0x438 (1080)	Blue Titanite slab
0x442 (1090)	White Titanite slab
0x44c (1100)	Red Titanite slab
0x456 (1110)	Dragon scales
0x460 (1120)	Demonic titanite
0x46a (1130)	Twinkling titanite
0x7d1 (2001)	Cellar Key
0x7d2 (2002)	Artorias symbol
0x7d3 (2003)	Cage Key
0x7d4 (2004)	Archive's prison key
	Stuff
0x7d5 (2005)	–The– Key
0x7d6 (2006)	–The– Key
0x7d7 (2007)	–The– Key
0x7d8 (2008)	–The– Key
0x7d9 (2009)	–The– Key
0x7da (2010)	–The– Key
0x7db (2011)	–The– Key
0x7dc (2012)	–The– Key
0x7dd (2013)	–The– Key
0x7de (2014)	–The– Key
0x7df (2015)	–The– Key
0x7e0 (2016)	–The– Key
0x7e1 (2017)	–The– Key
0x7e2 (2018)	–The– Key
0x7e3 (2019)	–The– Key

Continued on next page

Table 1.3 – continued from previous page

Item Type	Ingame Item title
0x7e4 (2020)	–The– Key
0x7e5 (2021)	–The– Key
0x7e6 (2022)	–The– Key
0x9c4 (2500)	The Great Soul
0x9c5 (2501)	The Great Soul #2
0x9c6 (2502)	Shard of great soul
0x9c7 (2503)	Shard of great soul #2
0x9ce (2510)	Lordvessel
0x9d8 (2520)	Broken neckle
0xa28 (2600)	Blacksmith's box
0xa29 (2601)	Blacksmith's armour box
0xa2a (2602)	Repair's box
0xa2b (2603)	Unknown Receipt #1
0xa2c (2604)	Unknown key #1
0xa2d (2605)	Unknown key #2
0xa2e (2606)	Unknown key #3
0xa2f (2607)	Rite of Kindling
0xa30 (2608)	Limitless box
0xa2f (2609)	Unknown key #4
	Sorcery
0xbb8 (3000)	Sorcery: soul arrow
0xbc2 (3010)	Sorcery: big soul arrow
0xbcc (3020)	Sorcery: heavy soul arrow
0xbd6 (3030)	Sorcery: huge soul arrow
0xbe0 (3040)	Sorcery: Homing Soulmass
0xbea (3050)	Sorcery: Crystal Soulmass
0xbf4 (3060)	Sorcery: Soul spear
0xbfe (3070)	Sorcery: Crystal soul spear
0xc1c (3100)	Magic weapon
0xc26 (3110)	Sorcery: big magic weapon
0xc30 (3120)	Sorcery: crystal magic weapon
0xce4 (3300)	Sorcery: magic shield
0xcee (3310)	Sorcery: strong magic shield
0xd48 (3400)	Sorcery: hidden weapon
0xd52 (3410)	Sorcery: hidden body
0xd52 (3500)	Sorcery: cast light
0xdb6 (3510)	Sorcery: hush
0xda (3530)	Sorcery: repair
0xdd4 (3540)	Sorcery: fall control
0xdde (3550)	Sorcery: chameleon
0xe10 (3600)	Sorcery: resist Curse
0xe1a (3610)	Sorcery: cure
0xe74 (3700)	Sorcery: breath of the white dragon
0xe7e (3710)	Sorcery: dark orb
0xe88 (3720)	Sorcery: dark bead
0xe92 (3730)	Sorcery: fog of darkness
0xe9 (3740)	Sorcery: Pursuers
	Pyromancy
0xfa0 (4000)	Pyromancy: Fire bead
0xfaa (4010)	Pyromancy: Fire orb
0xfb4 (4020)	Pyromancy: Big fire orb

Continued on next page

Table 1.3 – continued from previous page

Item Type	Ingame Item title
0xfbe (4030)	Pyromancy: Fire storm
0xfc8 (4040)	Pyromancy: Fire tempest
0xfd2 (4050)	Pyromancy: Fire surge
0xfdc (4060)	Pyromancy: Fire whip
0x1004 (4100)	Pyromancy: Combustion
0x100e (4110)	Pyromancy: Great combustion
0x1068 (4200)	Pyromancy: Poison mist
0x1072 (4210)	Pyromancy: Toxic mist
0x107c (4220)	Pyromancy: Acid surge
0x10cc (4300)	Pyromancy: Iron flesh
0x10d6 (4310)	Pyromancy: Flesh sweat
0x1130 (4400)	Pyromancy: Power within
0x1194 (4500)	Pyromancy: Big chaos orb
0x119e (4510)	Pyromancy: Chaos storm
0x11a8 (4520)	Pyromancy: Chaos fire whip
	Miracles (names of the spells not included yet)
0x1388 (5000)	Miracles: Heal
0x1388 (5010)	Miracles: Heal
0x1388 (5020)	Miracles: Heal
0x1388 (5030)	Miracles: Heal
0x1388 (5040)	Miracles: Heal
0x1388 (5050)	Miracles: Heal
0x1388 (5100)	Miracles: Heal
0x1388 (5110)	Miracles: Heal
0x1388 (5200)	Miracles: Heal
0x1388 (5210)	Miracles: Heal
0x1388 (5300)	Miracles: Heal
0x1388 (5310)	Miracles: Heal
0x1388 (5320)	Miracles: Heal
0x1388 (5400)	Miracles: Heal
0x1388 (5500)	Miracles: Heal
0x1388 (5510)	Miracles: Heal
0x1388 (5520)	Miracles: Heal
0x1388 (5600)	Miracles: Heal
0x1388 (5610)	Miracles: Heal
0x1388 (5700)	Miracles: Heal
0x1388 (5800)	Miracles: Heal
0x1388 (5810)	Miracles: Heal
0x1388 (5900)	Miracles: Heal
0x1388 (5910)	Miracles: Heal
	Emoticons (names are not included yet)
0x1388 (9000)	Emotions: Bow
0x1388 (9001)	Emotions: Bow
0x1388 (9002)	Emotions: Bow
0x1388 (9003)	Emotions: Bow
0x1388 (9004)	Emotions: Bow
0x1388 (9005)	Emotions: Bow
0x1388 (9006)	Emotions: Bow
0x1388 (9007)	Emotions: Bow
0x1388 (9008)	Emotions: Bow
0x1388 (9009)	Emotions: Bow

Continued on next page

Table 1.3 – continued from previous page

Item Type	Ingame Item title
0x1388 (9010)	Emotions: Bow
0x1388 (9011)	Emotions: Bow
0x1388 (9012)	Emotions: Bow
0x1388 (9013)	Emotions: Bow
0x1388 (9014)	Emotions: Bow
0x1388 (9015)	Emotions: Bow

1.5.2 Weapons store structure

```
from ctypes import
weapon_proto_structure = {'ffff': c_uint32, 'type': c_uint32, 'have', c_uint32, 'xxx': c_uint32}
```

Weapon Type	Ingame Item title
	Weapons
100000-100015	Dagger (up to +15)
103000-100015	Bandit's knife (up to +15)
852000-852005	Demonic Hummer (up to +5)

CHAPTER 2

dsfp — Darksouls Save File Parser

Note: Please note that this library not supported write routines and maybe won't in further releases. Only data get procedures

2.1 Core module

class dsfp.dsfp.**DSSaveFileParser**(*filename*)

Dark Souls save file parser original gist: https://gist.github.com/infuasto/8382836

Parameters **filename** (*str*) – bz2.BZ2File or StringIO instances

get_active_slots_amount()

get active slots count, could be 0 up to 9

Returns active characters' slots amount

Return type int

```
>>> instance = DSSaveFileParser('saves/DRAKS0005.sl2')
>>> instance.get_active_slots_amount()
2 # means that only 2 active characters stored in save file
```

get_blocks_metadata(*update=False*)

Get save file blocks metadata

Parameters **update** (*bool*) – runs re-read blocks metadata process, if False returns back cached list

Returns `SlotHeaderStructure` instances

Return type list

get_items(*slot=0*)

get character's item list

Parameters **slot** – character save slot (0 up to 9)

Returns list of dicts

get_stats()

get character stats data

Returns dicts

Return type list

```
>>> ds = DSSaveFileParser('saves/DRAKS0005.sl2')
>>> ds.get_stats()[0]
{'attunement': 11, 'body': 0, 'class': 2, 'color': 7,
 'deaths': 155, 'dexterity': 21, 'earned': 4037210,
 'endurance': 54, 'face': 1, 'faith': 9, 'gift': 1, 'hairs': 5,
 'hp': 1100, 'hp2': 1100, 'hp_current': 1100, 'humanity': 2,
 'intelligence': 11, 'level': 115, 'male': True,
 'name': u'\u041a\u0430\u0440\u043b', 'resistance': 12,
 'souls': 39848, 'stamina': 160, 'stamina2': 160, 'stamina3': 160,
 'strength': 50, 'time': 218834, 'vitality': 30}]
```

read_slot_data(*slot=0*)
read raw data of given slot

Parameters **slot** (*int*) – character slot

Returns character block bytes

Return type str

class dsfp.dsfp.**SlotHeaderStructure**
Characters containers slots header structure

2.2 Utils module

DSFP utils for some routines

dsfp.utils.**get_structure_fmt**(*structure_class*)
get structure format for further struct extraction

Parameters **structure_class** – ctypes Structure instances

Returns tuple of (format_string, block_size)

```
>>> class SomeStructure(Structure):
    _fields_ = [
        ('data', c_ulong), # 8 bytes
        ('amount', c_uint32) # 4 bytes
    ]
>>> get_format_fmt(SomeStructure)
('QI', 12)
```

2.3 Exceptions

DSFP utils for some routines

exception dsfp.exceptions.**FileTypeException**(**args*)
not a DarkSouls save file exception

exception dsfp.exceptions.**ImproperlyConfigured**(**args*)
Improperly configured exception

CHAPTER 3

tests — Unit tests documentation

Here represented documents about tests covered agains dsfp

3.1 DSFP

DSFP core file

class tests.reader.**TestDSFPReader**(*methodName='runTest'*)
DSFP parser reader unit tests

test_read_ds_file()
test get character dark souls file slots

test_read_ds_file_metadata()
read file metadata

class tests.utils.**TestUtils**(*methodName='runTest'*)
Utils tests

test_get_structure_fmt()
test utils.get_structure_fmt routine function

CHAPTER 4

Save files examples

4.1 Save files

4.1.1 Where can I find my save file?

Your Dark Souls save files represented as DRAKS0005.sl2 files and stored in `Documents/NBGI/DarkSouls/[your name]`.

4.1.2 What is DRAKS0005.sl2.bz2?

DRAKS0005.sl.bz2 is a compressed bzip2 file and stored in `saves` directory. It has 3 profiles:

- - NG+
- - Start Game
- Smithy - Start Game

This file stores as bzip2 compressed to maintain small repo size, please unpack it if you want to play with it in Dark Souls (PC version)

CHAPTER 5

Installation

For general installation you would probably need virtual environment with pip installed:

```
user@localhost$ virtualenv --no-site-packages venv
user@localhost$ source ve/bin/activate
user@localhost$ pip install -r requirements/base.txt
```

optional

```
user@localhost$ pip install -r requirements/docs.txt
```

5.1 Dependencies

- python 2.7+

CHAPTER 6

Tests

You could run tests via *python -m unittest module* or via *run_tests.sh* script

```
user@localhost$ ./run_tests.sh tests.TestDSFPReader
```

CHAPTER 7

Usage

You can parse Dark Souls save files and get data in standard python dictionary format

```
>>> import dsfp
>>> ds = dsfp.DSSaveFileParser('saves/DRAKS0005.sl2')
>>> data = ds.get_stats()
>>> data
[{
    'attunement': 8, 'body': 0, 'class': 0, 'color': 0, 'deaths': 0,
    'dexterity': 13, 'earned': 60, 'endurance': 12, 'face': 0, 'faith': 9,
    'gift': 0, 'hairs': 0, 'hp': 594, 'hp2': 594, 'hp_current': 594,
    'humanity': 0, 'intelligence': 9, 'level': 4,
    'male': False,  # False means female
    'name': u'TEST_2', 'resistance': 11, 'souls': 60, 'stamina': 95,
    'stamina2': 95, 'stamina3': 95, 'strength': 13,
    'time': 62, # in seconds
    'vitality': 11
   }]
```

that's all folks

CHAPTER 8

Indices and tables

- *genindex*
- *modindex*
- *search*

d

t