

## Zarządzanie czasem pracy

Generated by Doxygen 1.8.17



<b>1 Hierarchical Index</b>	<b>1</b>
1.1 Class Hierarchy	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 File Index</b>	<b>5</b>
3.1 File List	5
<b>4 Class Documentation</b>	<b>7</b>
4.1 Administrator Class Reference	7
4.1.1 Detailed Description	9
4.1.2 Constructor & Destructor Documentation	9
4.1.2.1 Administrator() [1/2]	9
4.1.2.2 Administrator() [2/2]	9
4.1.2.3 ~Administrator()	9
4.1.3 Member Function Documentation	9
4.1.3.1 dodaj_pracownikow_do_firmy()	9
4.1.3.2 dodajadministratoradofirmy()	9
4.1.3.3 dodajdofirmy()	9
4.1.3.4 dodajdofirmyadm()	10
4.1.3.5 dodajpracownikadofirmy()	10
4.1.3.6 drukujraportpracownikow()	10
4.1.3.7 stworzhaslo()	10
4.1.3.8 stworzlink()	11
4.1.3.9 stworzlogin()	11
4.1.3.10 wyswietl_administratora()	12
4.1.4 Member Data Documentation	12
4.1.4.1 link_firmy	12
4.1.4.2 listapracownikow	12
4.2 Dzien Class Reference	12
4.2.1 Detailed Description	13
4.2.2 Constructor & Destructor Documentation	13
4.2.2.1 Dzien() [1/2]	13
4.2.2.2 Dzien() [2/2]	13
4.2.2.3 ~Dzien()	13
4.2.3 Friends And Related Function Documentation	14
4.2.3.1 Administrator	14
4.2.3.2 Kalendarz	14
4.2.3.3 Pracownik	14
4.2.4 Member Data Documentation	14
4.2.4.1 dzien	14
4.2.4.2 miesiac	14

4.2.4.3 rok . . . . .	14
4.3 Firma Class Reference . . . . .	15
4.3.1 Detailed Description . . . . .	17
4.3.2 Constructor & Destructor Documentation . . . . .	17
4.3.2.1 Firma() [1/2] . . . . .	17
4.3.2.2 Firma() [2/2] . . . . .	17
4.3.2.3 ~Firma() . . . . .	17
4.3.3 Member Function Documentation . . . . .	17
4.3.3.1 czytajplikfirmy() . . . . .	17
4.3.3.2 sprawdzczyadministrator() . . . . .	18
4.3.3.3 sprawdzlinie() . . . . .	18
4.3.4 Friends And Related Function Documentation . . . . .	18
4.3.4.1 Logowanie . . . . .	18
4.3.5 Member Data Documentation . . . . .	18
4.3.5.1 Administratorzy . . . . .	18
4.3.5.2 link . . . . .	19
4.3.5.3 nazwa . . . . .	19
4.3.5.4 Pracownicy . . . . .	19
4.4 Haslo Class Reference . . . . .	19
4.4.1 Detailed Description . . . . .	19
4.4.2 Constructor & Destructor Documentation . . . . .	20
4.4.2.1 Haslo() [1/2] . . . . .	20
4.4.2.2 ~Haslo() . . . . .	20
4.4.2.3 Haslo() [2/2] . . . . .	20
4.4.3 Member Function Documentation . . . . .	20
4.4.3.1 nowehaslo() . . . . .	20
4.4.3.2 pokazefekt() . . . . .	20
4.4.4 Member Data Documentation . . . . .	20
4.4.4.1 dlugosc . . . . .	21
4.4.4.2 duzelitery . . . . .	21
4.4.4.3 ilosc . . . . .	21
4.4.4.4 malelitery . . . . .	21
4.4.4.5 symbole . . . . .	21
4.5 Kalendarz Class Reference . . . . .	21
4.5.1 Detailed Description . . . . .	22
4.5.2 Constructor & Destructor Documentation . . . . .	22
4.5.2.1 Kalendarz() . . . . .	22
4.5.2.2 ~Kalendarz() . . . . .	22
4.5.3 Member Function Documentation . . . . .	22
4.5.3.1 czy_przestepny() . . . . .	22
4.5.3.2 Ile_dni_ma_miesiac() . . . . .	22
4.5.3.3 jaki_dzien_tygodnia() . . . . .	22

4.5.3.4	ktory_dzien_w_roku()	23
4.5.3.5	wypisz_miesiac()	23
4.6	Lista_administratorow Class Reference	23
4.6.1	Detailed Description	25
4.6.2	Constructor & Destructor Documentation	25
4.6.2.1	Lista_administratorow()	25
4.6.2.2	~Lista_administratorow()	25
4.6.3	Member Function Documentation	25
4.6.3.1	czytajplikpracownikow()	25
4.6.3.2	dodaj_administradora()	26
4.6.3.3	dodaj_podwladnych()	26
4.6.3.4	pokazlisteadministratorow()	26
4.6.3.5	sprawdzdate()	26
4.6.3.6	sprawdzlinie()	26
4.6.3.7	uzupelnijdanepracownicze()	27
4.6.3.8	znajdz_pracownika()	27
4.6.4	Member Data Documentation	27
4.6.4.1	next	27
4.7	Lista_firm Class Reference	28
4.7.1	Detailed Description	30
4.7.2	Constructor & Destructor Documentation	30
4.7.2.1	Lista_firm() [1/2]	30
4.7.2.2	Lista_firm() [2/2]	30
4.7.2.3	~Lista_firm()	30
4.7.3	Member Function Documentation	30
4.7.3.1	czytajplik()	30
4.7.3.2	dodaj_firme()	31
4.7.3.3	sprawdzlinie()	31
4.7.3.4	ta()	31
4.7.3.5	wyswietl_liste()	31
4.7.3.6	znajdz_firme()	31
4.7.4	Friends And Related Function Documentation	32
4.7.4.1	Zarzadzajacy	32
4.7.5	Member Data Documentation	32
4.7.5.1	nastepna	32
4.8	Listadni Class Reference	32
4.8.1	Detailed Description	33
4.8.2	Constructor & Destructor Documentation	33
4.8.2.1	Listadni() [1/2]	34
4.8.2.2	~Listadni()	34
4.8.2.3	Listadni() [2/2]	34
4.8.3	Member Function Documentation	34

4.8.3.1 dodaj_dzien()	34
4.8.3.2 nastepna()	34
4.8.3.3 operator+()	35
4.8.3.4 sprawdzdate()	35
4.8.3.5 wypisz_okres()	35
4.8.3.6 wypiszokres()	35
4.8.3.7 wyswietl_liste_dni()	35
4.8.4 Friends And Related Function Documentation	35
4.8.4.1 Administrator	35
4.8.4.2 Kalendarz	36
4.8.4.3 Pracownik	36
4.8.5 Member Data Documentation	36
4.8.5.1 next	36
4.9 Listapracownikow Class Reference	36
4.9.1 Detailed Description	37
4.9.2 Constructor & Destructor Documentation	38
4.9.2.1 Listapracownikow() [1/2]	38
4.9.2.2 Listapracownikow() [2/2]	38
4.9.2.3 ~Listapracownikow()	38
4.9.3 Member Function Documentation	38
4.9.3.1 czytajplikpracownikow()	38
4.9.3.2 dodaj_pracownikow()	38
4.9.3.3 nastpna()	39
4.9.3.4 pokazlistepracownikow()	39
4.9.3.5 sprawdzdate()	39
4.9.3.6 sprawdzlinie()	39
4.9.3.7 uzupełnijdanepracownicze()	39
4.9.3.8 znajdz_pracownika()	40
4.9.3.9 znajdz_pracownika_adm()	40
4.9.4 Friends And Related Function Documentation	40
4.9.4.1 Administrator	40
4.9.5 Member Data Documentation	40
4.9.5.1 nastepny	41
4.10 Logowanie Class Reference	41
4.10.1 Constructor & Destructor Documentation	43
4.10.1.1 Logowanie() [1/2]	43
4.10.1.2 Logowanie() [2/2]	43
4.10.1.3 ~Logowanie()	43
4.10.2 Member Function Documentation	43
4.10.2.1 instrukcjaADM()	44
4.10.2.2 instrukcjaPRA()	44
4.10.2.3 logowanie()	44

4.10.2.4 sprawdzdate()	44
4.10.2.5 stworz_zarzadce()	44
4.10.3 Friends And Related Function Documentation	44
4.10.3.1 Zaradzajacy	44
4.10.4 Member Data Documentation	44
4.10.4.1 firma	45
4.10.4.2 haslo	45
4.10.4.3 login	45
4.10.4.4 tmp	45
4.11 Pracownik Class Reference	45
4.11.1 Detailed Description	46
4.11.2 Constructor & Destructor Documentation	46
4.11.2.1 Pracownik() [1/3]	47
4.11.2.2 Pracownik() [2/3]	47
4.11.2.3 Pracownik() [3/3]	47
4.11.2.4 ~Pracownik()	47
4.11.3 Member Function Documentation	47
4.11.3.1 dnipracy()	47
4.11.3.2 sprawdzokres()	48
4.11.3.3 stworzplikpracownika()	48
4.11.3.4 wpisz_do_bazy()	48
4.11.3.5 wyswietl_pracownika()	48
4.11.3.6 zamelduj_w_pracy()	48
4.11.4 Member Data Documentation	48
4.11.4.1 haslo	48
4.11.4.2 imie	49
4.11.4.3 link	49
4.11.4.4 lista	49
4.11.4.5 login	49
4.11.4.6 nazwisko	49
4.11.4.7 stanowisko	49
4.12 Zaradzajacy Class Reference	50
4.12.1 Detailed Description	52
4.12.2 Constructor & Destructor Documentation	52
4.12.2.1 Zaradzajacy()	52
4.12.2.2 ~Zaradzajacy()	52
4.12.3 Member Function Documentation	52
4.12.3.1 dodaj_firme()	52
4.12.3.2 dodaj_zaradzajacego()	52
4.12.3.3 dodajfirmedospisu()	53
4.12.3.4 sprawdzczyzaradzajacy()	53
4.12.3.5 sprawdzlinie()	53

4.12.4 Member Data Documentation . . . . .	53
4.12.4.1 haslo . . . . .	53
4.12.4.2 linkdospisu . . . . .	53
4.12.4.3 listafirm . . . . .	53
4.12.4.4 login . . . . .	53
<b>5 File Documentation</b>	<b>55</b>
5.1 Administrator.h File Reference . . . . .	55
5.2 Dzien.h File Reference . . . . .	57
5.3 Firma.h File Reference . . . . .	57
5.4 Haslo.h File Reference . . . . .	58
5.5 Kalendarz.h File Reference . . . . .	60
5.6 Lista_administratorow.h File Reference . . . . .	60
5.7 Lista_firm.h File Reference . . . . .	61
5.8 Lista_pracownikow.h File Reference . . . . .	62
5.9 Listadni.h File Reference . . . . .	63
5.10 Logowanie.h File Reference . . . . .	65
5.11 Pracownik.h File Reference . . . . .	65
5.12 Zaradzajacy.h File Reference . . . . .	66
<b>Index</b>	<b>69</b>



# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Dzien . . . . .	12
Listadni . . . . .	32
Firma . . . . .	15
Lista_firm . . . . .	28
Logowanie . . . . .	41
Haslo . . . . .	19
Kalendarz . . . . .	21
Pracownik . . . . .	45
Administrator . . . . .	7
Lista_administratorow . . . . .	23
Zaradzajacy . . . . .	50
Listapracownikow . . . . .	36



## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<b>Administrator</b>	7
<b>Dzien</b>	12
<b>Firma</b>	15
<b>Haslo</b>	19
<b>Kalendarz</b>	21
<b>Lista_administratorow</b>	23
<b>Lista_firm</b>	28
<b>Listadni</b>	32
<b>Listapracownikow</b>	36
<b>Logowanie</b>	41
<b>Pracownik</b>	45
<b>Zarzadzajacy</b>	50



## Chapter 3

# File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

<b>Administrator.h</b>	55
<b>Dzien.h</b>	57
<b>Firma.h</b>	57
<b>Haslo.h</b>	58
<b>Kalendarz.h</b>	60
<b>Lista_administratorow.h</b>	60
<b>Lista_firm.h</b>	61
<b>Lista_pracownikow.h</b>	62
<b>Listadni.h</b>	63
<b>Logowanie.h</b>	65
<b>Pracownik.h</b>	65
<b>Zaradzajacy.h</b>	66



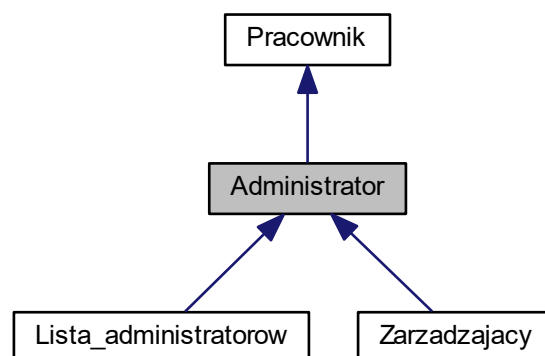
## Chapter 4

# Class Documentation

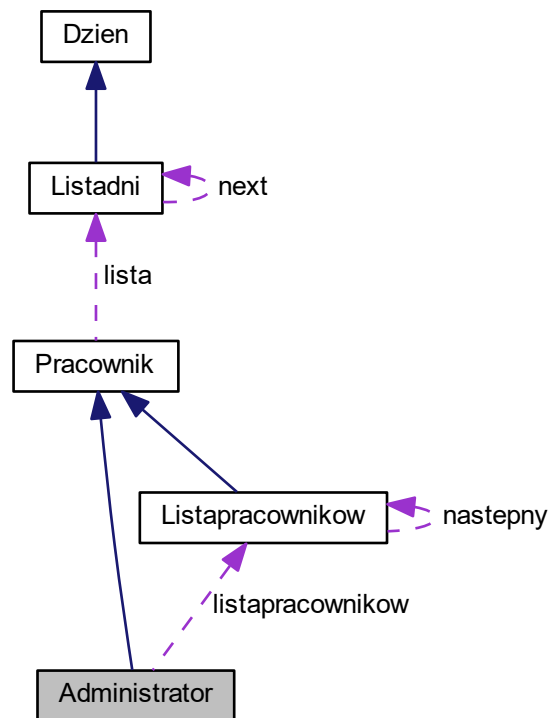
### 4.1 Administrator Class Reference

```
#include <Administrator.h>
```

Inheritance diagram for Administrator:



Collaboration diagram for Administrator:



## Public Member Functions

- **Administrator** (std::string sciezka)
- **Administrator** ()
- void **wyswietl\_administratora** ()
- void **dodajpracownikadofirmy** ()
- std::string **stworzlogin** (std::string imie, std::string nazwisko)
- std::string **stworzhaslo** ()
- std::string **stworzlink** (std::string login)
- void **dodajdofirmy** (std::string linkp, std::string linkf)
- void **dodaj\_pracownikow\_do\_firmy** ( Listapracownikow \*obiekt)
- void **drukujraportpracownikow** (int od\_d, int od\_m, int od\_r, int do\_d, int do\_m, int do\_r)
- virtual ~**Administrator** ()
- void **dodajadministratoradofirmy** (std::string linkf, std::string nazwaf)
- void **dodajdofirmyadm** (std::string linkp, std::string linkf, std::string nazwaf)

## Protected Attributes

- std::string **link\_firmy**
- Listapracownikow \* **listapracownikow**



### 4.1.1 Detailed Description

Klasa administrator jest jedna z 2 klas pracownikow w firmie. Klasa administrator posiada wieksze mozliwosci od klasy pracownik moze drukowac raporty , sprawdzac dni pracy innym pracownikow firmy ,dodawac nowych pracownikow.

### 4.1.2 Constructor & Destructor Documentation

#### 4.1.2.1 Administrator() [1/2]

```
Administrator::Administrator (
    std::string sciezka ) [inline]
```

#### 4.1.2.2 Administrator() [2/2]

```
Administrator::Administrator ( ) [inline]
```

#### 4.1.2.3 ~Administrator()

```
virtual Administrator::~~Administrator ( ) [inline], [virtual]
```

Virtualny destruktor.

### 4.1.3 Member Function Documentation

#### 4.1.3.1 dodaj\_pracownikow\_do\_firmy()

```
void Administrator::dodaj_pracownikow_do_firmy (
    Listapracownikow * obiekt )
```

Metoda dodaje obiekt do listypracownikow firmy. @param obiekt nowo powstaly pracownik

#### 4.1.3.2 dodajadministratoradofirmy()

```
void Administrator::dodajadministratoradofirmy (
    std::string linkf,
    std::string nazwaf )
```

#### 4.1.3.3 dodajdofirmy()

```
void Administrator::dodajdofirmy (
    std::string linkp,
    std::string linkf )
```

Metoda dodaje link do bazy danych pracownika do bazy danych firmy.

## Parameters

<i>linkp,link</i>	pracownika
<i>linkf,link</i>	firmy

**4.1.3.4 dodajdofirmyadm()**

```
void Administrator::dodajdofirmyadm (
    std::string linkp,
    std::string linkf,
    std::string nazwa )
```

**4.1.3.5 dodajpracownikadofirmy()**

```
void Administrator::dodajpracownikadofirmy ( )
```

Metoda sluzzy do utworzenia nowego pracownika i dodanie go do bazy danych firmy oraz listy pracownikow firmy.

**4.1.3.6 drukujraportpracownikow()**

```
void Administrator::drukujraportpracownikow (
    int od_d,
    int od_m,
    int od_r,
    int do_d,
    int do_m,
    int do_r )
```

Metoda drukuje raport wszystkich pracownikow firmy z danego okresu.

**4.1.3.7 stworzhaslo()**

```
std::string Administrator::stworzhaslo ( )
```

Metoda tworzy personalne haslo dla nowego pracownika

## Returns

haslo

#### 4.1.3.8 stworzlink()

```
std::string Administrator::stworzlink (
    std::string login )
```

Metoda tworzy personalny link do bazy danych pracownika dla nowego pracownika.

#### 4.1.3.9 stworzlogin()

```
std::string Administrator::stworzlogin (
    std::string imie,
    std::string nazwisko )
```

Metoda tworzy personalny login dla nowego pracownika.

**Parameters**

<i>imie</i>	
<i>nazwisko</i>	

**Returns**

login

**4.1.3.10 wyswietl\_administratora()**

```
void Administrator::wyswietl_administratora ( )
```

**4.1.4 Member Data Documentation****4.1.4.1 link\_firmy**

```
std::string Administrator::link_firmy [protected]
```

**4.1.4.2 listapracownikow**

```
Listapracownikow* Administrator::listapracownikow [protected]
```

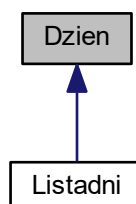
The documentation for this class was generated from the following file:

- **Administrator.h**

**4.2 Dzien Class Reference**

```
#include <Dzien.h>
```

Inheritance diagram for Dzien:



## Public Member Functions

- **Dzien** ()
- **Dzien** (int d, int m, int r)
- virtual **~Dzien** ()

## Protected Attributes

- int **dzien**
- int **miesiac**
- int **rok**

## Friends

- class **Pracownik**
- class **Kalendarz**
- class **Administrator**

### 4.2.1 Detailed Description

Klasa sluzy do przechowywania danych takich jak daty, czyli dnia w roku.

### 4.2.2 Constructor & Destructor Documentation

#### 4.2.2.1 Dzien() [1/2]

```
Dzien::Dzien ( ) [inline]
```

#### 4.2.2.2 Dzien() [2/2]

```
Dzien::Dzien (
    int d,
    int m,
    int r ) [inline]
```

#### 4.2.2.3 ~Dzien()

```
virtual Dzien::~~Dzien ( ) [inline], [virtual]
```

## 4.2.3 Friends And Related Function Documentation

### 4.2.3.1 Administrator

```
friend class Administrator [friend]
```

### 4.2.3.2 Kalendarz

```
friend class Kalendarz [friend]
```

### 4.2.3.3 Pracownik

```
friend class Pracownik [friend]
```

## 4.2.4 Member Data Documentation

### 4.2.4.1 dzien

```
int Dzien::dzien [protected]
```

### 4.2.4.2 miesiac

```
int Dzien::miesiac [protected]
```

### 4.2.4.3 rok

```
int Dzien::rok [protected]
```

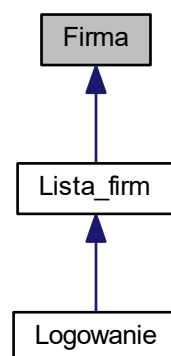
The documentation for this class was generated from the following file:

- **Dzien.h**

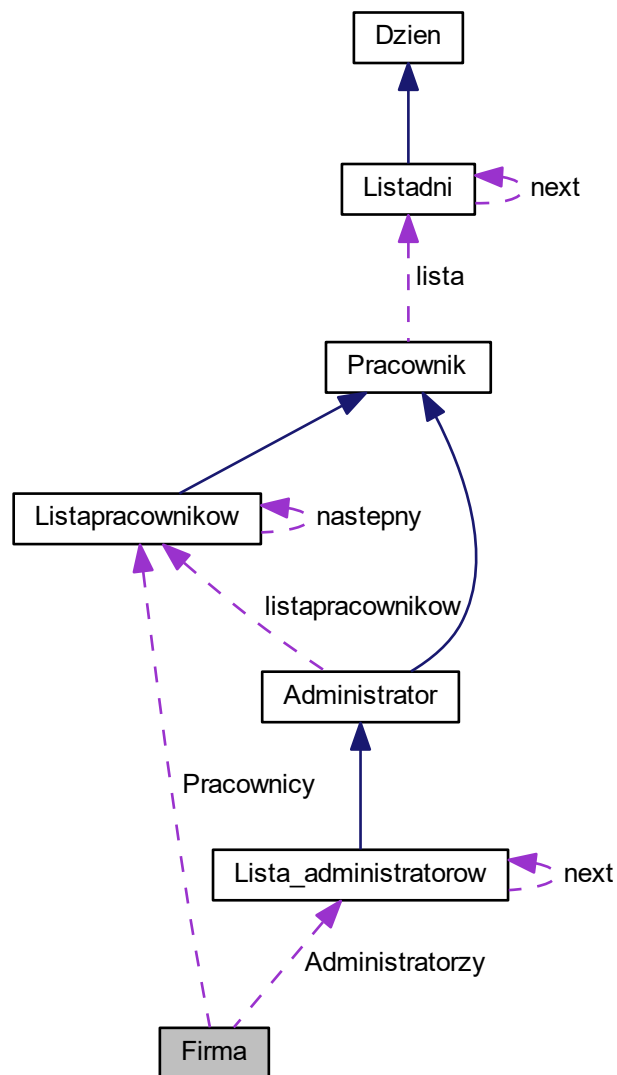
## 4.3 Firma Class Reference

```
#include <Firma.h>
```

Inheritance diagram for Firma:



Collaboration diagram for Firma:



## Public Member Functions

- **Firma** (std::string dopliku)
- **Firma** ()
- void **czytajplikfirmy** (std::string)
- bool **sprawdzlinie** (std::string napis)
- bool **sprawdzczyadministrator** (std::string link)
- virtual **~Firma** ()

## Protected Attributes

- std::string **link**



- `std::string nazwa`
- `Listapracownikow * Pracownicy`
- `Lista_administratorow * Administratorzy`

## Friends

- class `Logowanie`

### 4.3.1 Detailed Description

Klasa firma odpowiada rzeczywistemu obiektowi firma.

### 4.3.2 Constructor & Destructor Documentation

#### 4.3.2.1 Firma() [1/2]

```
Firma::Firma (
    std::string dopliku ) [inline]
```

#### 4.3.2.2 Firma() [2/2]

```
Firma::Firma ( ) [inline]
```

#### 4.3.2.3 ~Firma()

```
virtual Firma::~Firma ( ) [inline], [virtual]
```

Destruktro wirtualny.

### 4.3.3 Member Function Documentation

#### 4.3.3.1 czytajplikfirmy()

```
void Firma::czytajplikfirmy (
    std::string )
```

Metoda czyta plik firmy i tworzy listy pracownikow oraz administratorow.

**Parameters**

<i>link</i>	do pliku firmy
-------------	----------------

**4.3.3.2 sprawdzczyadministrator()**

```
bool Firma::sprawdzczyadministrator (
    std::string link )
```

Sprawdzenie czy dany link odpowiada za link do bazy danych administratora czy zwykłego pracownika.

**Parameters**

<i>link</i>	
-------------	--

**4.3.3.3 sprawdzlinie()**

```
bool Firma::sprawdzlinie (
    std::string napis )
```

Metoda sprawdza czy linki do baz danych mają poprawny format.

**4.3.4 Friends And Related Function Documentation****4.3.4.1 Logowanie**

```
friend class Logowanie [friend]
```

**4.3.5 Member Data Documentation****4.3.5.1 Administratorzy**

```
Lista_administratorow* Firma::Administratorzy [protected]
```

#### 4.3.5.2 link

```
std::string Firma::link [protected]
```

#### 4.3.5.3 nazwa

```
std::string Firma::nazwa [protected]
```

#### 4.3.5.4 Pracownicy

```
Listapracownikow* Firma::Pracownicy [protected]
```

The documentation for this class was generated from the following file:

- **Firma.h**

## 4.4 Haslo Class Reference

```
#include <Haslo.h>
```

### Public Member Functions

- **Haslo** ()
- **~Haslo** ()
- **Haslo** (int len, int amt, bool smA, bool lgA, bool sym)
- std::string **nowehaslo** (bool outMeta)
- void **pokazefekt** ()

### Protected Attributes

- int **dlugosc**
- int **ilosc**
- bool **malelitery**
- bool **duzelitery**
- bool **symbole**

#### 4.4.1 Detailed Description

Klasa haslo tworzy skomplikowane haslo.

## 4.4.2 Constructor & Destructor Documentation

### 4.4.2.1 Haslo() [1/2]

```
Haslo::Haslo ( )
```

### 4.4.2.2 ~Haslo()

```
Haslo::~~Haslo ( )
```

### 4.4.2.3 Haslo() [2/2]

```
Haslo::Haslo (
    int len,
    int amt,
    bool smA,
    bool lgA,
    bool sym )
```

## 4.4.3 Member Function Documentation

### 4.4.3.1 nowehaslo()

```
std::string Haslo::nowehaslo (
    bool outMeta )
```

### 4.4.3.2 pokazefekt()

```
void Haslo::pokazefekt ( )
```

## 4.4.4 Member Data Documentation

#### 4.4.4.1 dlugosc

```
int Haslo::dlugosc [protected]
```

#### 4.4.4.2 duzelitery

```
bool Haslo::duzelitery [protected]
```

#### 4.4.4.3 ilosc

```
int Haslo::ilosc [protected]
```

#### 4.4.4.4 malelitery

```
bool Haslo::malelitery [protected]
```

#### 4.4.4.5 symbole

```
bool Haslo::symbole [protected]
```

The documentation for this class was generated from the following file:

- **Haslo.h**

## 4.5 Kalendarz Class Reference

```
#include <Kalendarz.h>
```

### Public Member Functions

- **Kalendarz** ( **Listadni** \*tmp)
- **~Kalendarz** ()
- bool **czy\_przestepny** (int r)
- int **ile\_dni\_ma\_miesiac** (int m, int r)
- int **ktory\_dzien\_w\_roku** (int d, int m, int r)
- int **jaki\_dzien\_tygodnia** (int d, int m, int r)
- void **wypisz\_miesiac** (int m, int r, **Listadni** \*&tmp)

### 4.5.1 Detailed Description

Klasa kalendarz wypisuje dni pracy z danego okresu w formie kalendarza

### 4.5.2 Constructor & Destructor Documentation

#### 4.5.2.1 Kalendarz()

```
Kalendarz::Kalendarz (
    Listadni * tmp ) [inline]
```

#### 4.5.2.2 ~Kalendarz()

```
Kalendarz::~~Kalendarz ( ) [inline]
```

### 4.5.3 Member Function Documentation

#### 4.5.3.1 czy\_przestepny()

```
bool Kalendarz::czy_przestepny (
    int r )
```

Metoda sprawdza, czy rok r jest przestepny.

#### 4.5.3.2 Ile\_dni\_ma\_miesiac()

```
int Kalendarz::Ile_dni_ma_miesiac (
    int m,
    int r )
```

Metoda zwraca liczbe dni miesiaca m roku r.

#### 4.5.3.3 jaki\_dzien\_tygodnia()

```
int Kalendarz::jaki_dzien_tygodnia (
    int d,
    int m,
    int r )
```

Metoda oblicza, jaki dzien tygodnia przedstawia data d, m, r.

#### 4.5.3.4 ktory\_dzien\_w\_roku()

```
int Kalendarz::ktory_dzien_w_roku (
    int d,
    int m,
    int r )
```

Metoda oblicza, ktory z kolei dzien w roku przedstawia data d, m, r.

#### 4.5.3.5 wypisz\_miesiac()

```
void Kalendarz::wypisz_miesiac (
    int m,
    int r,
    Listadni *& tmp )
```

Metoda wypisuje dni pracy danego pracownika w formie kalendarza

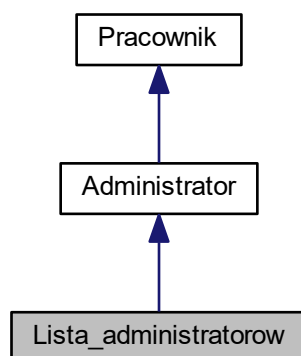
The documentation for this class was generated from the following file:

- Kalendarz.h

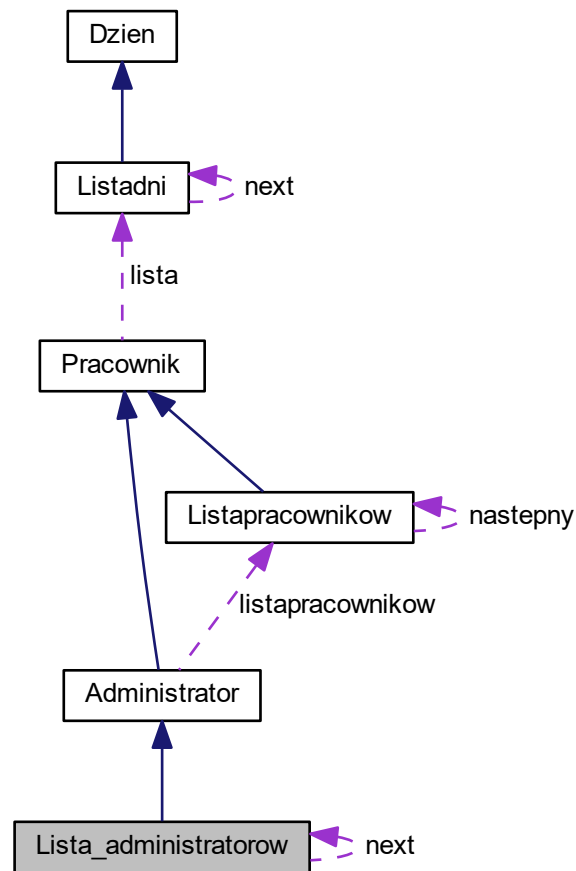
## 4.6 Lista\_administratorow Class Reference

```
#include <Lista_administratorow.h>
```

Inheritance diagram for Lista\_administratorow:



Collaboration diagram for Lista\_administratorow:



## Public Member Functions

- **Lista\_administratorow** ()
- void **czytajplikpracownikow** (std::string sciezka)
- void **dodaj\_administradora** (std::string sciezka)
- bool **sprawdzlinie** (std::string linia)
- void **pokazlisteadministratorow** ()
- void **uzupelnijdanepracownicze** (std::string linia)
- **Administrator \*** **znajdz\_pracownika** (std::string login, std::string haslo)
- bool **sprawdzdate** (int dzien, int miesiac, int rok)
- void **dodaj\_podwladnych** ( **Listapracownikow \***tmp, std::string link\_f)
- **~Lista\_administratorow** ()

## Protected Attributes

- **Lista\_administratorow \*** next



### 4.6.1 Detailed Description

Klasa spelnia obiekt listy administratorow

### 4.6.2 Constructor & Destructor Documentation

#### 4.6.2.1 Lista\_administratorow()

```
Lista_administratorow::Lista_administratorow ( ) [inline]
```

#### 4.6.2.2 ~Lista\_administratorow()

```
Lista_administratorow::~~Lista_administratorow ( ) [inline]
```

Here is the call graph for this function:



### 4.6.3 Member Function Documentation

#### 4.6.3.1 czytajplikpracownikow()

```
void Lista_administratorow::czytajplikpracownikow (
    std::string sciezka )
```

Czyta plik administratora .

##### Parameters

<i>link</i>	do bazy danych administratora
-------------	-------------------------------

#### 4.6.3.2 dodaj\_administratora()

```
void Lista_administratorow::dodaj_administratora (
    std::string sciezka )
```

Metoda dodaje administratora do listy administratorow.

##### Parameters

<i>sciezka</i>	
----------------	--

#### 4.6.3.3 dodaj\_podwladnych()

```
void Lista_administratorow::dodaj_podwladnych (
    Listapracownikow * tmp,
    std::string link_f )
```

Metoda dodaje wskaznik na liste pracownikow firmy do administratora.

#### 4.6.3.4 pokazlisteadministratorow()

```
void Lista_administratorow::pokazlisteadministratorow ( )
```

Here is the caller graph for this function:



#### 4.6.3.5 sprawdzdate()

```
bool Lista_administratorow::sprawdzdate (
    int dzien,
    int miesiac,
    int rok )
```

Sprawdz czy jest taki dzien w roku.

#### 4.6.3.6 sprawdzlinie()

```
bool Lista_administratorow::sprawdzlinie (
    std::string linia )
```

Sprawadza czy napis spelnia format linku do pliku.

## Parameters

<i>napis</i>	
--------------	--

#### 4.6.3.7 uzupełnijdanepracownicze()

```
void Lista_administratorow::uzupełnijdanepracownicze (
    std::string linia )
```

Uzupełnia dane pracownika o o dane i dni pracy

## Parameters

<i>linia,link</i>	do bazy danych
-------------------	----------------

#### 4.6.3.8 znajdz\_pracownika()

```
Administrator* Lista_administratorow::znajdz_pracownika (
    std::string login,
    std::string haslo )
```

Metoda znajduje administratora po loginie i hasle w liscie administratorow.

## Returns

wskaznik na obiekt administrator

### 4.6.4 Member Data Documentation

#### 4.6.4.1 next

```
Lista_administratorow* Lista_administratorow::next [protected]
```

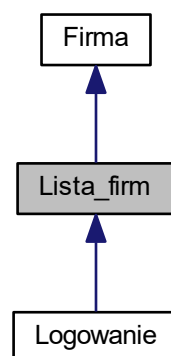
The documentation for this class was generated from the following file:

- **Lista\_administratorow.h**

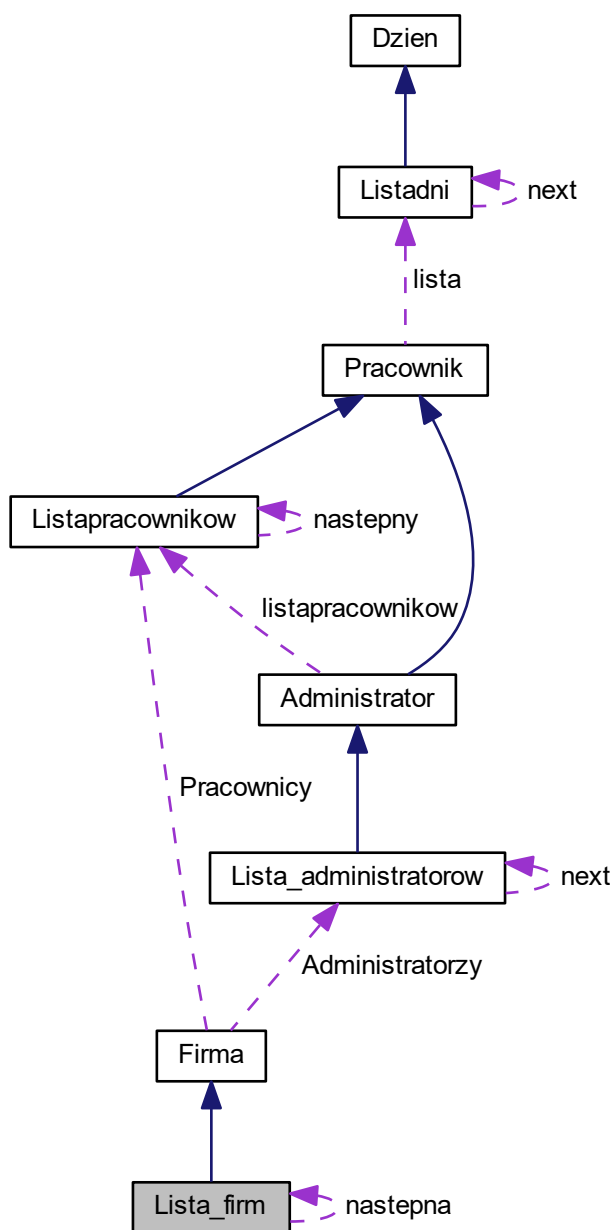
## 4.7 Lista\_firm Class Reference

```
#include <Lista_firm.h>
```

Inheritance diagram for Lista\_firm:



Collaboration diagram for Lista\_firm:



## Public Member Functions

- `Lista_firm ()`
- `Lista_firm (std::string link)`
- `void czytajplik (std::string nazwa)`
- `void dodaj_firme (std::string sciezka)`
- `bool sprawdzlinie (std::string napis)`
- `Lista_firm * znajdz_firme (std::string nazwa_firmy)`

- void `wyswietl_liste()`
- `~Lista_firm()`
- `Lista_firm * ta()`

## Protected Attributes

- `Lista_firm * nastepna`

## Friends

- class `Zarzadzajacy`

### 4.7.1 Detailed Description

Klasa tworzy liste firm

### 4.7.2 Constructor & Destructor Documentation

#### 4.7.2.1 `Lista_firm()` [1/2]

```
Lista_firm::Lista_firm ( ) [inline]
```

#### 4.7.2.2 `Lista_firm()` [2/2]

```
Lista_firm::Lista_firm (
    std::string link ) [inline]
```

#### 4.7.2.3 `~Lista_firm()`

```
Lista_firm::~~Lista_firm ( ) [inline]
```

### 4.7.3 Member Function Documentation

#### 4.7.3.1 `czytajplik()`

```
void Lista_firm::czytajplik (
    std::string nazwa )
```

Funkcja czyta plik z spisem firm i tworzy liste firm

## Parameters

<i>nazwa,nazwa</i>	pliku z dopiskiem .txt gdzie znajdują sie nazwy plikow kazdej z firm
--------------------	--

#### 4.7.3.2 dodaj\_firme()

```
void Lista_firm::dodaj_firme (
    std::string sciezka )
```

Funkcja tworzy liste firm i uzupełni dane kazdej z firm

## Parameters

<i>sciezka,nazwa</i>	pliku firmy
----------------------	-------------

#### 4.7.3.3 sprawdzlinie()

```
bool Lista_firm::sprawdzlinie (
    std::string napis )
```

#### 4.7.3.4 ta()

```
Lista_firm* Lista_firm::ta ( ) [inline]
```

#### 4.7.3.5 wyswietl\_liste()

```
void Lista_firm::wyswietl_liste ( )
```

#### 4.7.3.6 znajdz\_firme()

```
Lista_firm* Lista_firm::znajdz_firme (
    std::string nazwa_firmy )
```

Funkcja szuka firmy o danej nazwie i zwraca wskaznik na nia

#### Parameters

<code>nazwa_firmy,nazwa</code>	<code>firmy ,ktorej szukamy</code>
--------------------------------	------------------------------------

#### Returns

wskaznik do szukanego obiektu

### 4.7.4 Friends And Related Function Documentation

#### 4.7.4.1 Zarzadzajacy

```
friend class Zarzadzajacy [friend]
```

### 4.7.5 Member Data Documentation

#### 4.7.5.1 nastepna

```
Lista_firm* Lista_firm::nastepna [protected]
```

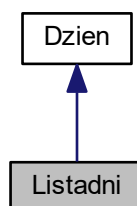
The documentation for this class was generated from the following file:

- **Lista\_firm.h**

## 4.8 Listadni Class Reference

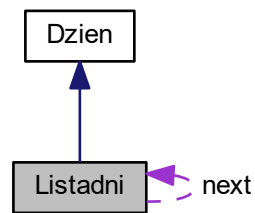
```
#include <Listadni.h>
```

Inheritance diagram for Listadni:





Collaboration diagram for Listadni:



## Public Member Functions

- **Listadni** ()
- bool **dodaj\_dzien** (int, int, int)
- void **wyświetl\_liste\_dni** ()
- bool **sprawdzdate** (int **dzien**, int **miesiac**, int **rok**)
- **Listadni & operator+** ( **Listadni** \*&tmp)
- void **wypisz\_okres** ()
- void **wypiszokres** (int od\_d, int od\_m, int od\_rok, int do\_d, int do\_m, int do\_rok)
- **~Listadni** ()
- **Listadni** (const **Listadni** \*&tmp)
- **Listadni** \* **nastepna** ()

## Protected Attributes

- **Listadni** \* **next**

## Friends

- class **Pracownik**
- class **Kalendarz**
- class **Administrator**

### 4.8.1 Detailed Description

Klasa odpowiadająca za tworzenie listy dni

### 4.8.2 Constructor & Destructor Documentation

#### 4.8.2.1 Listadni() [1/2]

```
Listadni::Listadni ( ) [inline]
```

#### 4.8.2.2 ~Listadni()

```
Listadni::~~Listadni ( ) [inline]
```

#### 4.8.2.3 Listadni() [2/2]

```
Listadni::Listadni (
    const Listadni *& tmp ) [inline]
```

### 4.8.3 Member Function Documentation

#### 4.8.3.1 dodaj\_dzien()

```
bool Listadni::dodaj_dzien (
    int ,
    int ,
    int )
```

Metoda dodaje dzien do listy dni

##### Returns

true ,jesli dzien sie nie potworzyl  
false gdy jest juz taki dzien

#### 4.8.3.2 nastepna()

```
Listadni* Listadni::nastepna ( ) [inline]
```

#### 4.8.3.3 operator+()

```
Listadni& Listadni::operator+ (
    Listadni *& tmp )
```

Operator doadaje dzien do liste dni

#### 4.8.3.4 sprawdzdate()

```
bool Listadni::sprawdzdate (
    int dzien,
    int miesiac,
    int rok )
```

#### 4.8.3.5 wypisz\_okres()

```
void Listadni::wypisz_okres ( )
```

W tej metodzie tworzy sie i wypisuje kalendarz.

#### 4.8.3.6 wypiszokres()

```
void Listadni::wypiszokres (
    int od_d,
    int od_m,
    int od_rok,
    int do_d,
    int do_m,
    int do_rok )
```

Metoda tworzy liste dni spelniajacych warunki

#### 4.8.3.7 wyswietl\_liste\_dni()

```
void Listadni::wyswietl_liste_dni ( )
```

### 4.8.4 Friends And Related Function Documentation

#### 4.8.4.1 Administrator

```
friend class Administrator [friend]
```

#### 4.8.4.2 Kalendarz

```
friend class Kalendarz [friend]
```

#### 4.8.4.3 Pracownik

```
friend class Pracownik [friend]
```

### 4.8.5 Member Data Documentation

#### 4.8.5.1 next

```
Listadni* Listadni::next [protected]
```

The documentation for this class was generated from the following file:

- Listadni.h

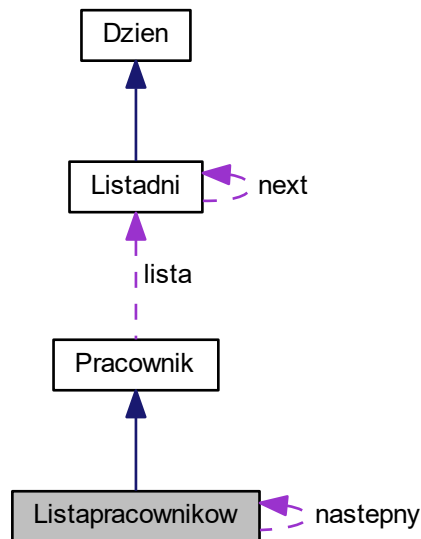
## 4.9 Listapracownikow Class Reference

```
#include <Lista_pracownikow.h>
```

Inheritance diagram for Listapracownikow:



Collaboration diagram for Listpracownikow:



## Public Member Functions

- **Listpracownikow** ()
- **Listpracownikow** (std::string h, std::string l, std::string i, std::string n, std::string s, std::string li)
- void **czytajplikpracownikow** (std::string sciezka)
- void **dodaj\_pracownikow** (std::string sciezka)
- bool **sprawdzlinie** (std::string linia)
- void **pokazlistepracownikow** ()
- void **uzupelnijdanepracownicze** (std::string linia)
- **Pracownik \* znajdź\_pracownika** (std::string login, std::string haslo)
- bool **sprawdzdate** (int dzien, int miesiac, int rok)
- **Pracownik \* znajdź\_pracownika\_adm** (std::string imie, std::string nazwisko)
- virtual **~Listpracownikow** ()
- **Listpracownikow \* nastpna** ()

## Protected Attributes

- **Listpracownikow \* nastepny**

## Friends

- class **Administrator**

### 4.9.1 Detailed Description

Klasa jest jest lista pracownikow

## 4.9.2 Constructor & Destructor Documentation

### 4.9.2.1 Listapracownikow() [1/2]

```
Listapracownikow::Listapracownikow ( ) [inline]
```

### 4.9.2.2 Listapracownikow() [2/2]

```
Listapracownikow::Listapracownikow (
    std::string h,
    std::string l,
    std::string i,
    std::string n,
    std::string s,
    std::string li ) [inline]
```

### 4.9.2.3 ~Listapracownikow()

```
virtual Listapracownikow::~~Listapracownikow ( ) [inline], [virtual]
```

## 4.9.3 Member Function Documentation

### 4.9.3.1 czytajplikpracownikow()

```
void Listapracownikow::czytajplikpracownikow (
    std::string sciezka )
```

### 4.9.3.2 dodaj\_pracownikow()

```
void Listapracownikow::dodaj_pracownikow (
    std::string sciezka )
```

Funkcja tworzy liste pracownikow danej firmy

## Parameters

<i>sciezka,link</i>	do pliku pracownika
---------------------	---------------------

#### 4.9.3.3 nastpna()

```
Listapracownik* Listapracownik::nastpna ( ) [inline]
```

#### 4.9.3.4 pokazlistepracownikow()

```
void Listapracownik::pokazlistepracownikow ( )
```

#### 4.9.3.5 sprawdzdate()

```
bool Listapracownik::sprawdzdate (
    int dzien,
    int miesiac,
    int rok )
```

#### 4.9.3.6 sprawdzlinie()

```
bool Listapracownik::sprawdzlinie (
    std::string linia )
```

Funkcja sprawdza linie wczytana z pliku

## Parameters

<i>napis,linia</i>	wczytana z pliku
--------------------	------------------

## Returns

true ,gdy w pobranej lini 4 ostatnie znaki to .txt

#### 4.9.3.7 uzupełnijdanepracownicze()

```
void Listapracownik::uzupełnijdanepracownicze (
    std::string linia )
```

Funkcja uzupełnia dane pracownikow

#### Parameters

<i>nazwa</i>	linker do pliku ,w którym zapisane sa dane pracownika
--------------	---

#### 4.9.3.8 `znajdz_pracownika()`

```
Pracownik* ListaPracownikow::znajdz_pracownika (
    std::string login,
    std::string haslo )
```

Metoda szuka pracownika po loginie i hasle i zwraca wskaznik na obiekt

#### Returns

wskaznik na obiekt **Pracownik** (p. 45)

#### 4.9.3.9 `znajdz_pracownika_adm()`

```
Pracownik* ListaPracownikow::znajdz_pracownika_adm (
    std::string imie,
    std::string nazwisko )
```

Metoda szuka pracownika po imieniu i nazwisku i zwraca wskaznik na obiekt.

#### Returns

wskaznik na obiekt **Pracownik** (p. 45)

### 4.9.4 Friends And Related Function Documentation

#### 4.9.4.1 Administrator

```
friend class Administrator [friend]
```

### 4.9.5 Member Data Documentation



#### 4.9.5.1 następny

```
Listapracownik* Listapracownik::nastepny [protected]
```

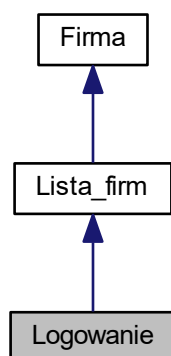
The documentation for this class was generated from the following file:

- Lista\_pracownikow.h

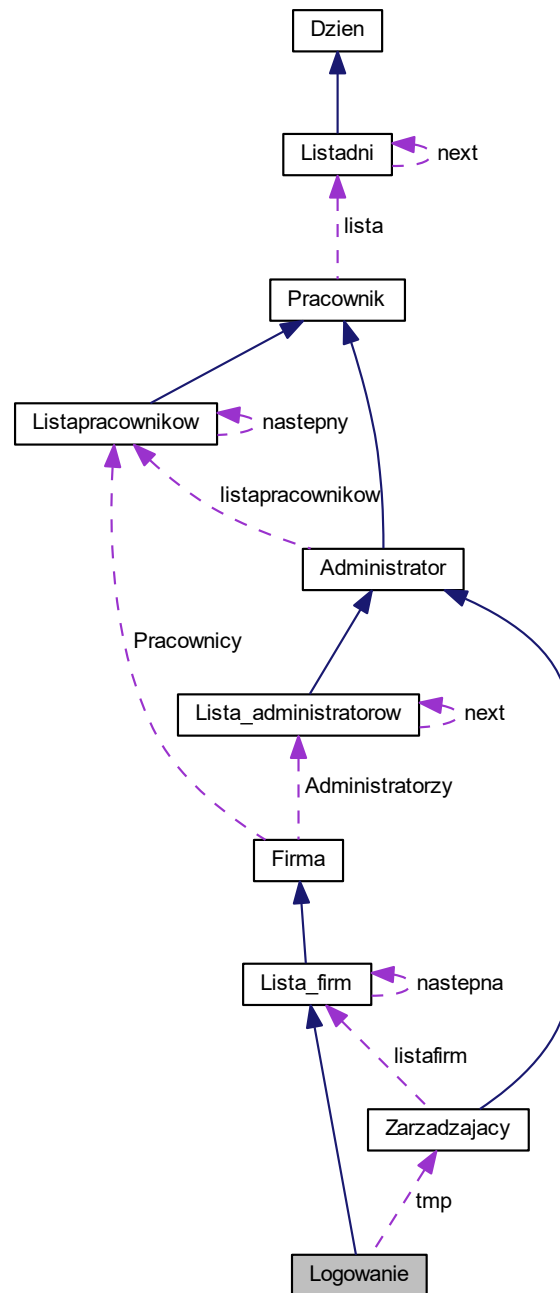
## 4.10 Logowanie Class Reference

```
#include <Logowanie.h>
```

Inheritance diagram for Logowanie:



Collaboration diagram for Logowanie:



## Public Member Functions

- **Logowanie** (std::string **login**, std::string **haslo**, std::string **firma**)
- **Logowanie** ()
- void **logowanie** ()
- void **instrukcjaADM** ()
- void **instrukcjaPRA** ()

- bool **sprawdzdate** (int dzien, int miesiac, int rok)
- **~Logowanie** ()
- void **stworz\_zarzadce** (std::string spis, **Lista\_firm** \*&lista)

### Public Attributes

- std::string **login** = ""
- std::string **haslo** = ""
- std::string **firma** = ""

### Protected Attributes

- **Zarzadzajacy** \* tmp

### Friends

- class **Zarzadzajacy**

## 4.10.1 Constructor & Destructor Documentation

### 4.10.1.1 Logowanie() [1/2]

```
Logowanie::Logowanie (
    std::string login,
    std::string haslo,
    std::string firma ) [inline]
```

### 4.10.1.2 Logowanie() [2/2]

```
Logowanie::Logowanie ( ) [inline]
```

### 4.10.1.3 ~Logowanie()

```
Logowanie::~~Logowanie ( ) [inline]
```

## 4.10.2 Member Function Documentation

#### 4.10.2.1 instrukcjaADM()

```
void Logowanie::instrukcjaADM ( )
```

Wypisuje insturkcje dla administratра.

#### 4.10.2.2 instrukcjaPRA()

```
void Logowanie::instrukcjaPRA ( )
```

Wypisuje insturkcje dla pracownika.

#### 4.10.2.3 logowanie()

```
void Logowanie::logowanie ( )
```

Metoda sterujaca wszystkimi dzialaniami ,czesc integralna programu.

#### 4.10.2.4 sprawdzdate()

```
bool Logowanie::sprawdzdate (
    int dzien,
    int miesiac,
    int rok )
```

#### 4.10.2.5 stworz\_zarzadce()

```
void Logowanie::stworz_zarzadce (
    std::string spis,
    Lista_firm *& lista )
```

### 4.10.3 Friends And Related Function Documentation

#### 4.10.3.1 Zarzadzajacy

```
friend class  Zarzadzajacy  [friend]
```

### 4.10.4 Member Data Documentation

#### 4.10.4.1 firma

```
std::string Logowanie::firma = ""
```

#### 4.10.4.2 haslo

```
std::string Logowanie::haslo = ""
```

#### 4.10.4.3 login

```
std::string Logowanie::login = ""
```

#### 4.10.4.4 tmp

```
Zarzadzajacy* Logowanie::tmp [protected]
```

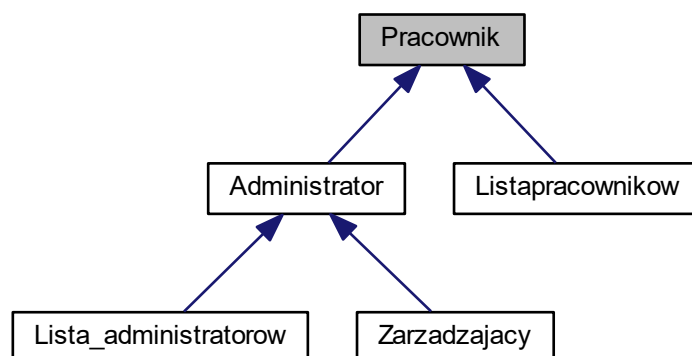
The documentation for this class was generated from the following file:

- **Logowanie.h**

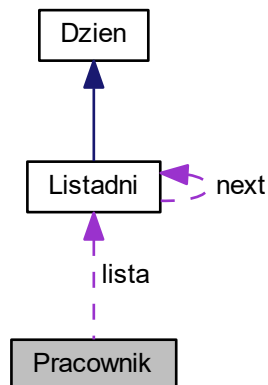
## 4.11 Pracownik Class Reference

```
#include <Pracownik.h>
```

Inheritance diagram for Pracownik:



Collaboration diagram for Pracownik:



## Public Member Functions

- **Pracownik** (std::string h, std::string l, std::string i, std::string n, std::string s, std::string li)
- **Pracownik** (std::string sciezka)
- **Pracownik** ()
- void **zamelduj\_w\_pracy** ()
- void **wpisz\_do\_bazy** (const **Dzień** \*data)
- void **wyswietl\_pracownika** ()
- void **sprawdzokres** (int od\_d, int od\_m, int od\_rok, int do\_d, int do\_m, int do\_rok)
- **Listadni** \* **dnipracy** (int od\_d, int od\_m, int od\_rok, int do\_d, int do\_m, int do\_rok)
- void **stworzplikpracownika** ()
- virtual **~Pracownik** ()

## Protected Attributes

- std::string **haslo**
- std::string **login**
- std::string **imie**
- std::string **nazwisko**
- std::string **stanowisko**
- std::string **link**
- **Listadni** \* **lista**

### 4.11.1 Detailed Description

Klasa jest rzeczywistym obiektem pracownika .

### 4.11.2 Constructor & Destructor Documentation

#### 4.11.2.1 Pracownik() [1/3]

```
Pracownik::Pracownik (
    std::string h,
    std::string l,
    std::string i,
    std::string n,
    std::string s,
    std::string li ) [inline]
```

#### 4.11.2.2 Pracownik() [2/3]

```
Pracownik::Pracownik (
    std::string sciezka ) [inline]
```

#### 4.11.2.3 Pracownik() [3/3]

```
Pracownik::Pracownik ( ) [inline]
```

#### 4.11.2.4 ~Pracownik()

```
virtual Pracownik::~~Pracownik ( ) [inline], [virtual]
```

### 4.11.3 Member Function Documentation

#### 4.11.3.1 dnipracy()

```
Listadni* Pracownik::dnipracy (
    int od_d,
    int od_m,
    int od_rok,
    int do_d,
    int do_m,
    int do_rok )
```

Metoda zwraca liste dni spelniajacych warunki

##### Returns

wskaznik na liste dni

#### 4.11.3.2 sprawdzokres()

```
void Pracownik::sprawdzokres (
    int od_d,
    int od_m,
    int od_rok,
    int do_d,
    int do_m,
    int do_rok )
```

#### 4.11.3.3 stworzplikpracownika()

```
void Pracownik::stworzplikpracownika ( )
```

Tworzy plik nowego pracownika

#### 4.11.3.4 wpisz\_do\_bazy()

```
void Pracownik::wpisz_do_bazy (
    const Dzien * data )
```

Wpisuje dzien do bazy danych pracownika

#### 4.11.3.5 wyswietl\_pracownika()

```
void Pracownik::wyswietl_pracownika ( )
```

#### 4.11.3.6 zamelduj\_w\_pracy()

```
void Pracownik::zamelduj_w_pracy ( )
```

Metoda rejestruje przybycie do pracy w danym dni

### 4.11.4 Member Data Documentation

#### 4.11.4.1 haslo

```
std::string Pracownik::haslo [protected]
```



#### 4.11.4.2 imie

`std::string Pracownik::imie [protected]`

#### 4.11.4.3 link

`std::string Pracownik::link [protected]`

#### 4.11.4.4 lista

`Listadni* Pracownik::lista [protected]`

#### 4.11.4.5 login

`std::string Pracownik::login [protected]`

#### 4.11.4.6 nazwisko

`std::string Pracownik::nazwisko [protected]`

#### 4.11.4.7 stanowisko

`std::string Pracownik::stanowisko [protected]`

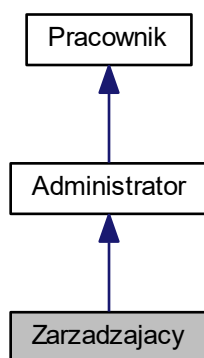
The documentation for this class was generated from the following file:

- **Pracownik.h**

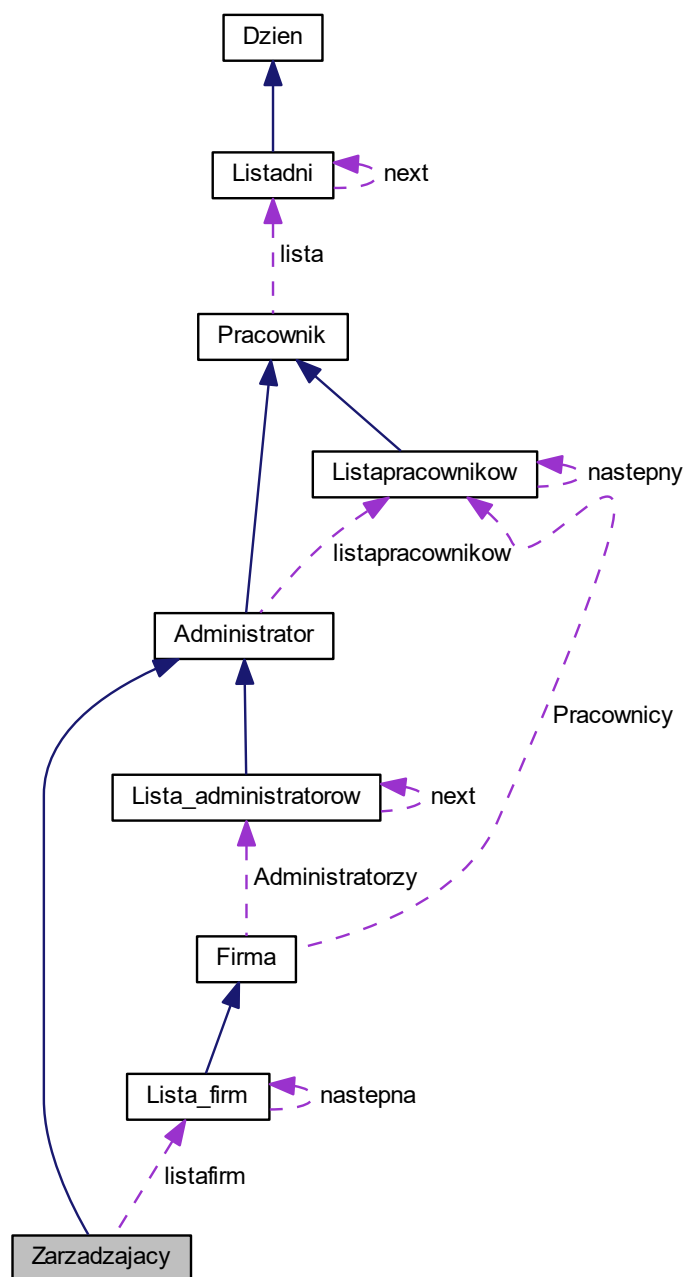
## 4.12 Zaradzajacy Class Reference

```
#include <Zaradzajacy.h>
```

Inheritance diagram for Zaradzajacy:



Collaboration diagram for Zarządzający:



## Public Member Functions

- **Zarządzający** ()
- **~Zarządzający** ()
- bool **sprawdzlinie** (std::string)
- void **dodaj\_zarządzającego** (std::string link, Lista\_firm \*& lista)
- void **dodaj\_firme** ()

- bool **sprawdzczyzarzadzajacy** (std::string h, std::string l)
- void **dodajfirmedospisu** ( **Lista\_firm** \*tmp)

## Protected Attributes

- **Lista\_firm** \* listafirm
- std::string **login**
- std::string **haslo**
- std::string **linkdospisu**

### 4.12.1 Detailed Description

Klasa z ktorej mozna dodawac firmy

### 4.12.2 Constructor & Destructor Documentation

#### 4.12.2.1 Zarzadzajacy()

```
Zarzadzajacy::Zarzadzajacy ( ) [inline]
```

#### 4.12.2.2 ~Zarzadzajacy()

```
Zarzadzajacy::~~Zarzadzajacy ( ) [inline]
```

### 4.12.3 Member Function Documentation

#### 4.12.3.1 dodaj\_firme()

```
void Zarzadzajacy::dodaj_firme ( )
```

#### 4.12.3.2 dodaj\_zarzadzajacego()

```
void Zarzadzajacy::dodaj_zarzadzajacego (
    std::string link,
    Lista_firm *& lista )
```

#### 4.12.3.3 dodajfirmedospisu()

```
void Zarzadzajacy::dodajfirmedospisu (
    Lista_firm * tmp )
```

#### 4.12.3.4 sprawdzczyzarzadzajacy()

```
bool Zarzadzajacy::sprawdzczyzarzadzajacy (
    std::string h,
    std::string l )
```

#### 4.12.3.5 sprawdzlinie()

```
bool Zarzadzajacy::sprawdzlinie (
    std::string )
```

### 4.12.4 Member Data Documentation

#### 4.12.4.1 haslo

```
std::string Zarzadzajacy::haslo [protected]
```

#### 4.12.4.2 linkdospisu

```
std::string Zarzadzajacy::linkdospisu [protected]
```

#### 4.12.4.3 listafirm

```
Lista_firm* Zarzadzajacy::listafirm [protected]
```

#### 4.12.4.4 login

```
std::string Zarzadzajacy::login [protected]
```

The documentation for this class was generated from the following file:

- **Zarzadzajacy.h**



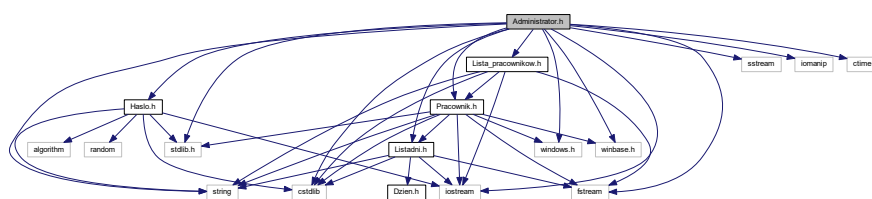
## Chapter 5

# File Documentation

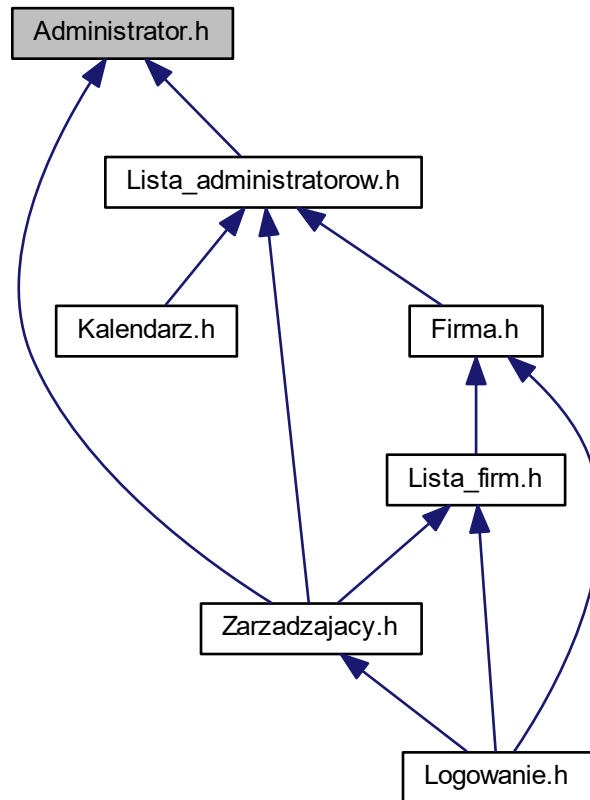
### 5.1 Administrator.h File Reference

```
#include <cstdlib>
#include <string>
#include <iostream>
#include <fstream>
#include <windows.h>
#include <winbase.h>
#include <stdlib.h>
#include <sstream>
#include <iomanip>
#include <ctime>
#include "Listadni.h"
#include "Pracownik.h"
#include "Haslo.h"
#include "Lista_pracownikow.h"
```

Include dependency graph for Administrator.h:



This graph shows which files directly or indirectly include this file:



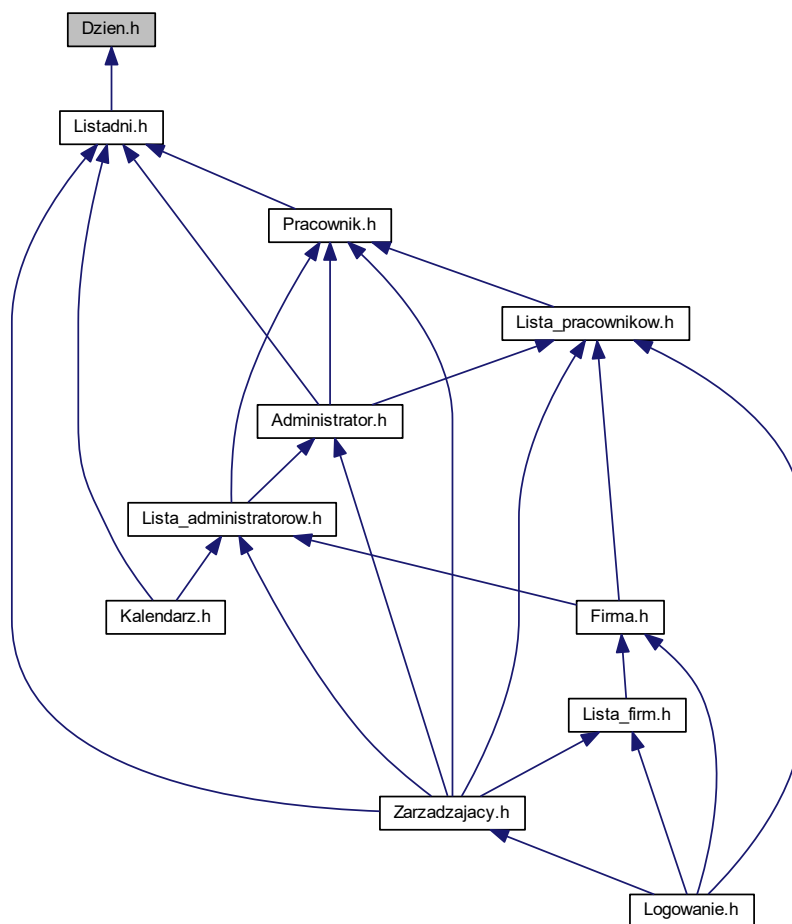
## Classes

- class **Administrator**



## 5.2 Dzień.h File Reference

This graph shows which files directly or indirectly include this file:



### Classes

- class **Dzien**

## 5.3 Firma.h File Reference

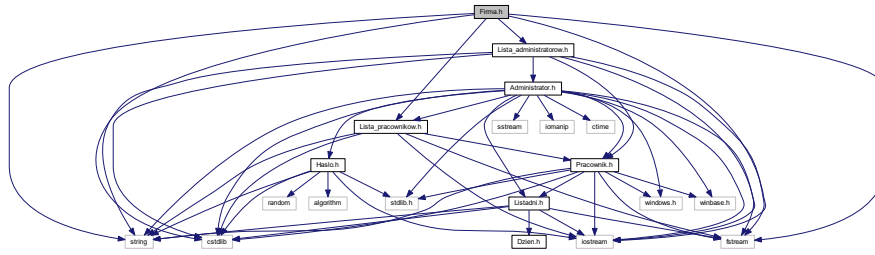
```

#include <cstdlib>
#include <string>
#include <iostream>
#include <fstream>
#include "Lista_pracownikow.h"

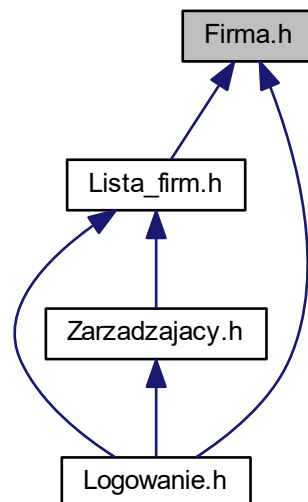
```

```
#include "Lista_administratorow.h"
```

Include dependency graph for Firma.h:



This graph shows which files directly or indirectly include this file:



## Classes

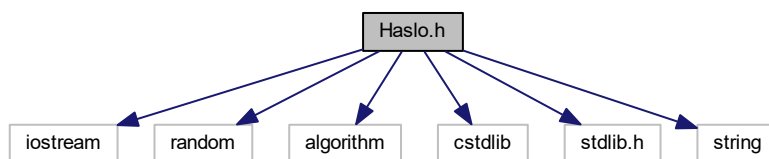
- class **Firma**

## 5.4 Haslo.h File Reference

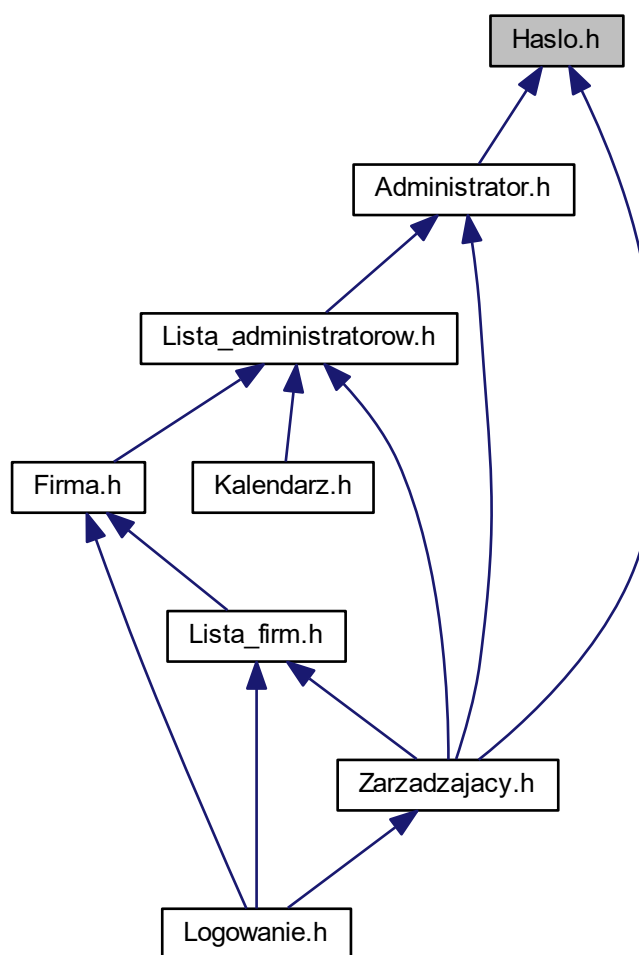
```
#include <iostream>
#include <random>
#include <algorithm>
#include <cstdlib>
#include <stdlib.h>
```

```
#include <string>
```

Include dependency graph for Haslo.h:



This graph shows which files directly or indirectly include this file:

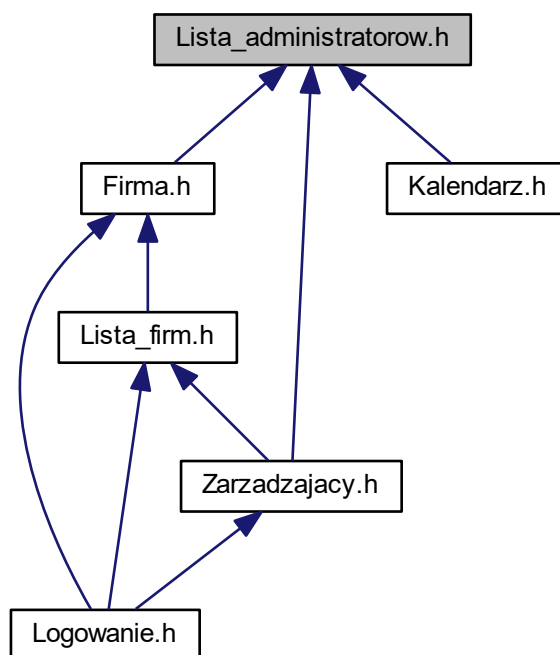


## Classes

- class **Haslo**



This graph shows which files directly or indirectly include this file:

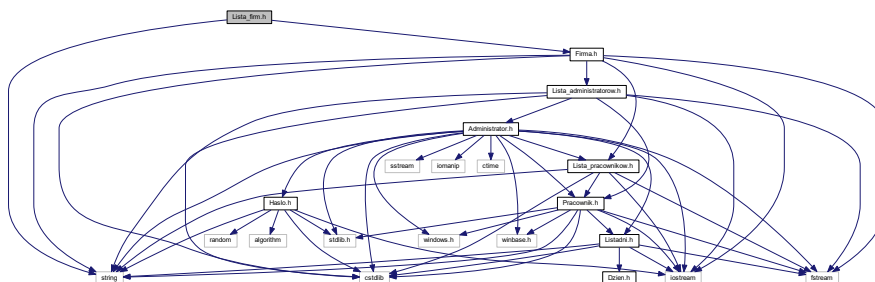


## Classes

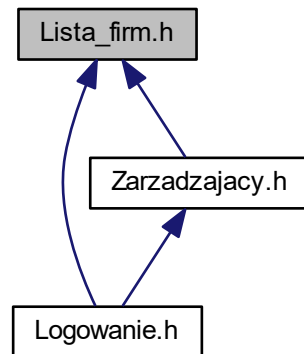
- class **Lista\_administratorow**

## 5.7 Lista\_firm.h File Reference

```
#include "Firma.h"
#include <string>
Include dependency graph for Lista_firm.h:
```



This graph shows which files directly or indirectly include this file:



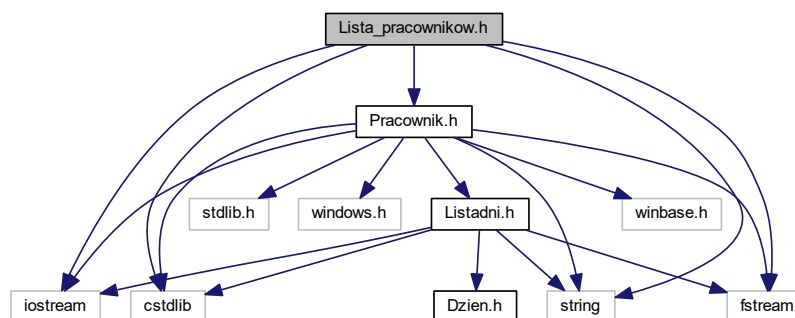
## Classes

- class `Lista_firm`

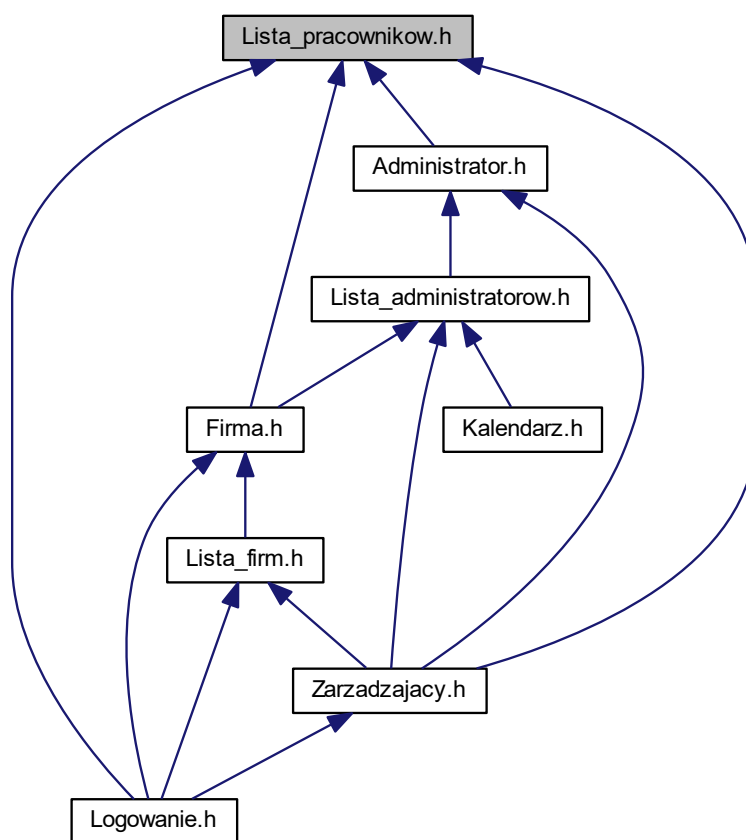
## 5.8 Lista\_pracownikow.h File Reference

```
#include <cstdlib>
#include <string>
#include <iostream>
#include <fstream>
#include "Pracownik.h"
```

Include dependency graph for `Lista_pracownikow.h`:



This graph shows which files directly or indirectly include this file:



## Classes

- class `Listapracownikow`

## 5.9 Listadni.h File Reference

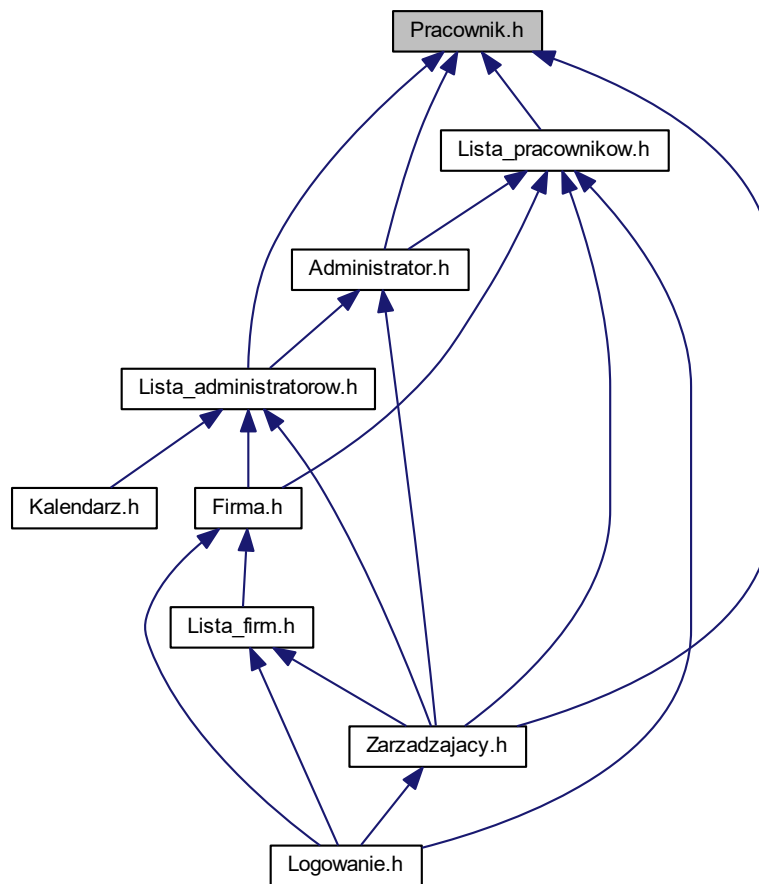
```
#include "Dzien.h"
#include <cstdlib>
#include <string>
#include <iostream>
#include <fstream>
```







This graph shows which files directly or indirectly include this file:



## Classes

- class **Pracownik**

## 5.12 Zarzadzajacy.h File Reference

```

#include <cstdlib>
#include <string>
#include <iostream>
#include <fstream>
#include <windows.h>
#include <winbase.h>
#include <stdlib.h>
#include <sstream>
#include <iomanip>
#include <ctime>
#include "Listadni.h"

```





# Index

- ~Administrator
  - Administrator, 9
- ~Dzien
  - Dzien, 13
- ~Firma
  - Firma, 17
- ~Haslo
  - Haslo, 20
- ~Kalendarz
  - Kalendarz, 22
- ~Lista\_administratorow
  - Lista\_administratorow, 25
- ~Lista\_firm
  - Lista\_firm, 30
- ~Listadni
  - Listadni, 34
- ~Listapracownikow
  - Listapracownikow, 38
- ~Logowanie
  - Logowanie, 43
- ~Pracownik
  - Pracownik, 47
- ~Zarzadzajacy
  - Zarzadzajacy, 52
- Administrator, 7
  - ~Administrator, 9
  - Administrator, 9
  - dodaj\_pracownikow\_do\_firmy, 9
  - dodajadministratoradofirmy, 9
  - dodajdofirmy, 9
  - dodajdofirmyadm, 10
  - dodajpracownikadofirmy, 10
  - drukujraportpracownikow, 10
  - Dzien, 14
  - link\_firmy, 12
  - Listadni, 35
  - Listapracownikow, 40
  - listapracownikow, 12
  - stworzhaslo, 10
  - stworzlink, 10
  - stworzlogin, 11
  - wyswietl\_administratora, 12
- Administrator.h, 55
- Administratorzy
  - Firma, 18
- czy\_przestepny
  - Kalendarz, 22
- czytajplik
  - Lista\_firm, 30
- czytajplikfirmy
  - Firma, 17
- czytajplikpracownikow
  - Lista\_administratorow, 25
  - Listapracownikow, 38
- dlugosc
  - Haslo, 20
- dnipracy
  - Pracownik, 47
- dodaj\_administradora
  - Lista\_administratorow, 25
- dodaj\_dzien
  - Listadni, 34
- dodaj\_firme
  - Lista\_firm, 31
  - Zarzadzajacy, 52
- dodaj\_podwladnych
  - Lista\_administratorow, 26
- dodaj\_pracownikow
  - Listapracownikow, 38
- dodaj\_pracownikow\_do\_firmy
  - Administrator, 9
- dodaj\_zarzadzajacego
  - Zarzadzajacy, 52
- dodajadministratoradofirmy
  - Administrator, 9
- dodajdofirmy
  - Administrator, 9
- dodajdofirmyadm
  - Administrator, 10
- dodajfirmedospisu
  - Zarzadzajacy, 52
- dodajpracownikadofirmy
  - Administrator, 10
- drukujraportpracownikow
  - Administrator, 10
- duzelitery
  - Haslo, 21
- Dzien, 12
  - ~Dzien, 13
  - Administrator, 14
  - Dzien, 13
  - dzien, 14
  - Kalendarz, 14
  - miesiac, 14
  - Pracownik, 14
  - rok, 14
- dzien

- Dzien, 14
- Dzien.h, 57
- Firma, 15
  - ~Firma, 17
  - Administratorzy, 18
  - czytajplikfirmy, 17
  - Firma, 17
  - link, 18
  - Logowanie, 18
  - nazwa, 19
  - Pracownicy, 19
  - sprawdzczyadministrator, 18
  - sprawdzlinie, 18
- firma
  - Logowanie, 44
- Firma.h, 57
- Haslo, 19
  - ~Haslo, 20
  - dlugosc, 20
  - duzelitery, 21
  - Haslo, 20
  - ilosc, 21
  - malelitery, 21
  - nowehaslo, 20
  - pokazefekt, 20
  - symbole, 21
- haslo
  - Logowanie, 45
  - Pracownik, 48
  - Zarzadzajacy, 53
- Haslo.h, 58
- Ile\_dni\_ma\_miesiac
  - Kalendarz, 22
- ilosc
  - Haslo, 21
- imie
  - Pracownik, 48
- instrukcjaADM
  - Logowanie, 43
- instrukcjaPRA
  - Logowanie, 44
- jaki\_dzien\_tygodnia
  - Kalendarz, 22
- Kalendarz, 21
  - ~Kalendarz, 22
  - czy\_przestepny, 22
  - Dzien, 14
  - Ile\_dni\_ma\_miesiac, 22
  - jaki\_dzien\_tygodnia, 22
  - Kalendarz, 22
  - ktory\_dzien\_w\_roku, 22
  - Listadni, 35
  - wypisz\_miesiac, 23
- Kalendarz.h, 60
- ktory\_dzien\_w\_roku
  - Kalendarz, 22
- link
  - Firma, 18
  - Pracownik, 49
- link\_firmy
  - Administrator, 12
- linkdospisu
  - Zarzadzajacy, 53
- lista
  - Pracownik, 49
- Lista\_administratorow, 23
  - ~Lista\_administratorow, 25
  - czytajplikpracownikow, 25
  - dodaj\_administradora, 25
  - dodaj\_podwladnych, 26
  - Lista\_administratorow, 25
  - next, 27
  - pokazlisteadministratorow, 26
  - sprawdzdate, 26
  - sprawdzlinie, 26
  - uzupelnijdanepracownicze, 27
  - znajdz\_pracownika, 27
- Lista\_administratorow.h, 60
- Lista\_firm, 28
  - ~Lista\_firm, 30
  - czytajplik, 30
  - dodaj\_firme, 31
  - Lista\_firm, 30
  - nastepna, 32
  - sprawdzlinie, 31
  - ta, 31
  - wyswietl\_liste, 31
  - Zarzadzajacy, 32
  - znajdz\_firme, 31
- Lista\_firm.h, 61
- Lista\_pracownikow.h, 62
- Listadni, 32
  - ~Listadni, 34
  - Administrator, 35
  - dodaj\_dzien, 34
  - Kalendarz, 35
  - Listadni, 33, 34
  - nastepna, 34
  - next, 36
  - operator+, 34
  - Pracownik, 36
  - sprawdzdate, 35
  - wypisz\_okres, 35
  - wypiszokres, 35
  - wyswietl\_liste\_dni, 35
- Listadni.h, 63
- listafirm
  - Zarzadzajacy, 53
- Listapracownikow, 36
  - ~Listapracownikow, 38
  - Administrator, 40
  - czytajplikpracownikow, 38

- dodaj\_pracownikow, 38
- Listapracownikow, 38
- nastepny, 40
- nastpna, 39
- pokazlistepracownikow, 39
- sprawdzdate, 39
- sprawdzlinie, 39
- uzupelnijdanepracownicze, 39
- znajdz\_pracownika, 40
- znajdz\_pracownika\_adm, 40
- listapracownikow
  - Administrator, 12
- login
  - Logowanie, 45
  - Pracownik, 49
  - Zaradzajacy, 53
- Logowanie, 41
  - ~Logowanie, 43
  - Firma, 18
  - firma, 44
  - haslo, 45
  - instrukcjaADM, 43
  - instrukcjaPRA, 44
  - login, 45
  - Logowanie, 43
  - logowanie, 44
  - sprawdzdate, 44
  - stworz\_zaradce, 44
  - tmp, 45
  - Zaradzajacy, 44
- logowanie
  - Logowanie, 44
- Logowanie.h, 65
- malelity
  - Haslo, 21
- miesiac
  - Dzien, 14
- nastepna
  - Lista\_firm, 32
  - Listadni, 34
- nastepny
  - Listapracownikow, 40
- nastpna
  - Listapracownikow, 39
- nazwa
  - Firma, 19
- nazwisko
  - Pracownik, 49
- next
  - Lista\_administratorow, 27
  - Listadni, 36
- nowehaslo
  - Haslo, 20
- operator+
  - Listadni, 34
- pokazefekt
  - Haslo, 20
- pokazlisteadministratorow
  - Lista\_administratorow, 26
- pokazlistepracownikow
  - Listapracownikow, 39
- Pracownicy
  - Firma, 19
- Pracownik, 45
  - ~Pracownik, 47
  - dnipracy, 47
  - Dzien, 14
  - haslo, 48
  - imie, 48
  - link, 49
  - lista, 49
  - Listadni, 36
  - login, 49
  - nazwisko, 49
  - Pracownik, 46, 47
  - sprawdzokres, 47
  - stanowisko, 49
  - stworzplikpracownika, 48
  - wpisz\_do\_bazy, 48
  - wyswietl\_pracownika, 48
  - zamelduj\_w\_pracy, 48
- Pracownik.h, 65
- rok
  - Dzien, 14
- sprawdzczyadministrator
  - Firma, 18
- sprawdzczyzaradzajacy
  - Zaradzajacy, 53
- sprawdzdate
  - Lista\_administratorow, 26
  - Listadni, 35
  - Listapracownikow, 39
  - Logowanie, 44
- sprawdzlinie
  - Firma, 18
  - Lista\_administratorow, 26
  - Lista\_firm, 31
  - Listapracownikow, 39
  - Zaradzajacy, 53
- sprawdzokres
  - Pracownik, 47
- stanowisko
  - Pracownik, 49
- stworz\_zaradce
  - Logowanie, 44
- stworzhaslo
  - Administrator, 10
- stworzlink
  - Administrator, 10
- stworzlogin
  - Administrator, 11
- stworzplikpracownika

- Pracownik, 48
- symbole
  - Haslo, 21
- ta
  - Lista\_firm, 31
- tmp
  - Logowanie, 45
- uzupelnijdanepracownicze
  - Lista\_administratorow, 27
  - Listapracownikow, 39
- wpisz\_do\_bazy
  - Pracownik, 48
- wypisz\_miesiac
  - Kalendarz, 23
- wypisz\_okres
  - Listadni, 35
- wypiszokres
  - Listadni, 35
- wyswietl\_administradora
  - Administrator, 12
- wyswietl\_liste
  - Lista\_firm, 31
- wyswietl\_liste\_dni
  - Listadni, 35
- wyswietl\_pracownika
  - Pracownik, 48
- zamelduj\_w\_pracy
  - Pracownik, 48
- Zarzadzajacy, 50
  - ~Zarzadzajacy, 52
  - dodaj\_firme, 52
  - dodaj\_zarzadzajacego, 52
  - dodajfirmedospisu, 52
  - haslo, 53
  - linkdospisu, 53
  - Lista\_firm, 32
  - listafirm, 53
  - login, 53
  - Logowanie, 44
  - sprawdzczyzarzadzajacy, 53
  - sprawdzlinie, 53
  - Zarzadzajacy, 52
- Zarzadzajacy.h, 66
- znajdz\_firme
  - Lista\_firm, 31
- znajdz\_pracownika
  - Lista\_administratorow, 27
  - Listapracownikow, 40
- znajdz\_pracownika\_adm
  - Listapracownikow, 40