



**CPCS 241**  
**Databases**  
**Group Project – Phase II**

**Student Names:** .....

**Student Numbers:** .....

**Group Number:** .....

**Semester:** Spring 2022

**Submission Date:** Monday 10/09/1443 H, **11/04/2022 G**

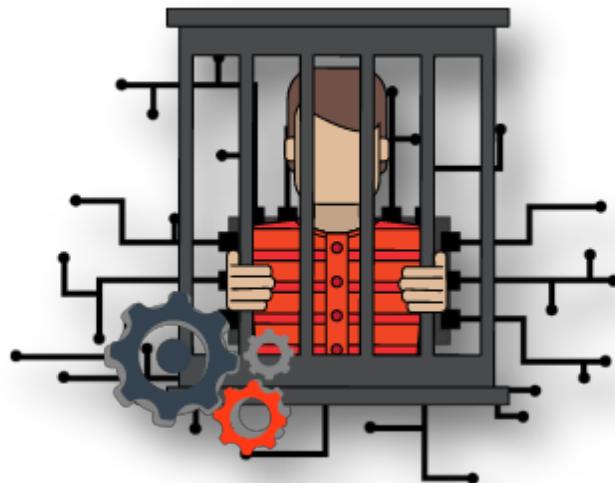
**Submission Time:** DAR, GAR & IAR @ 12:00 PM

**Submission Format:** Paper copy + Soft copy via BB

Criteria	Max Mark	Obtained Mark
Analysis	1	
ER Design (HVCLO6)	2	
ER-to-Relational Mapping (HVCLO10)	2	
Normalization (HVCLO11)	2	
<b>Total</b>	<b>7</b>	

Instructors: Dr. Lama Al Khuzayem and Dr. Rsha Mirza

[lalkhuzayem@kau.edu.sa](mailto:lalkhuzayem@kau.edu.sa) , [rmirza@kau.edu.sa](mailto:rmirza@kau.edu.sa)



# [City Jail Management System]

## DB Design

CPCS241 – Database I – Spring 2022 – Group Project

Group No: 11

Student Name	Student Number
Fai Badr Alnagrani	2006268
Shrooq Kaabi	2205003
Shahad Hatem Balkhasher	2006676
Majd Saleh	1816903
Wajd Alharbi	2007057

# Contents

PART I: Analysis .....	3
1 Problem Definition and Data Requirements.....	3
1.1 Problem Description .....	3
1.2 Data Requirements .....	4
1.3 Business Rules .....	7
1.4 Intended Output of the system.....	10
PART II: DB DESIGN .....	11
2 ER Diagram Design .....	11
2.1 ER diagram .....	11
2.2 Design of Business Rules .....	12
3 ER-to-logical schema mapping.....	16
3.1 Mapping of Regular Entity Types .....	16
3.2 Mapping of Weak Entity Types .....	20
3.3 Mapping of binary 1-1 relationship types.....	22
3.4 Mapping of binary 1-N relationship types .....	22
3.5 Mapping of binary M-N relationship types.....	27
3.6 Mapping of multivalued attributes .....	27
3.7 Mapping of n-ary relationship types.....	27
3.8 Schema Diagram .....	28
4 Normalization.....	29
4.1 First Normal Form .....	29
4.2 Second Normal Form .....	30
4.3 Third Normal Form.....	33
5 Final DB Schema Diagram .....	35

## PART I: Analysis

### 1 Problem Definition and Data Requirements

#### 1.1 Problem Description

As time has passed, the technological revolution has been used in the justice system to assist it. The utilization of a proper management system will without doubt be effective in improving data manipulation. To manage the city jail system, we designed a database to store, maintain, and access all the information about the prisoners detained inside the prison, which will enhance the way the prison is administrated, and help cities efficiently track and process inmates from the day they arrive to the day they are released. Every jail system consists of police officers (people in control) and prisoners (people under authority). In this system, prisoners are being watched and managed by police officers, creating a strict atmosphere that prevents prisoners from escaping or committing chaos.

## 1.2 Data Requirements

### 1. CELLS

- Each CELL has a unique cell number.
- Each CELL has a capacity (the maximum number of prisoners that can occupy a cell).
- Each CELL has a state (cell can be either occupied or unoccupied).

### 2. PRISONERS

- Each PRISONER has a prisoner number (a number that uniquely identifies the prisoner inside the prison).
- Each PRISONER has a composite attribute name (first name, last name).
- Each PRISONER has a date of arrival (the date on which the prisoner has arrived at the prison).
- Each PRISONER has a date of release (the date on which the prisoner is supposed to leave the prison).

### 3. HEALTH RECORDS

- HEALTH RECORDS is a weak entity of PRISONERS.
- Each HEALTH RECORD has the prisoner's name which is the partial key.
- Each HEALTH RECORD has the prisoner's height.
- Each HEALTH RECORD has the prisoner's weight.
- Each HEALTH RECORD has the prisoner's date of birth.
- Each HEALTH RECORD has the prisoner's state (single, married, widowed, etc.).
- Each HEALTH RECORD has the prisoner's blood type.
- Each HEALTH RECORD has the prisoner's disease history (a record of a prisoner's symptoms, illnesses, conditions, and alterations in development).

### 4. STAFF

- Each STAFF member has a staff ID.
- Each STAFF member has a composite attribute name (first name, last name).

- Each STAFF member has gender.
- Each STAFF member has a job type.
- Each STAFF member has a salary.
- Each STAFF member has an address.
- STAFF has specializations: {PRISON OFFICERS, YARD OFFICERS, HEAD DOCTOR, and PRISON COOK} based on the STAFF member's job type.

## 5. VISITS

- Each VISIT has the visitor's ID.
- Each VISIT has the visitor's name.
- Each VISIT has the visitor's address.
- Each VISIT has the visitor's phone number.
- Each VISIT has an ~~nmbr~~ of visits (Each prisoner has a certain number of visits).
- Each VISIT has the time of the visit (the time at which the visit happens).
- Each VISIT has the duration of the visit (the time during which the visit continues).

## 6. FLOOR

- Each FLOOR has a floor number.
- Each FLOOR has a floor's type (either cell's floor or warden's floor).

## 7. CELL'S FLOOR

- CELL'S FLOOR IS-A FLOOR that has the number of cells as an attribute.

## 8. WARDEN'S FLOOR

- WARDEN'S FLOOR IS-A FLOOR that has WARDEN'S office number as an attribute.

## 9. CRIMINAL RECORDS

- A CRIMINAL RECORD is a weak entity.
- Each CRIMINAL RECORD has the prisoner's name as a partial key.

- Each CRIMINAL RECORD has the prisoner's crimes.

## 10. ACTIVITIES

- PRISONERS can be assigned an ACTIVITY (a helpful activity).
- Each ACTIVITY has an activity number.
- Each ACTIVITY has an activity type (cleaning, laundry, community work, etc.).
- Each ACTIVITY has a reward (payment, early discharge).

## 11. PUNISHMENT

- It is a result of violating the law or committing any crime inside the prison.
- Each PUNISHMENT has a Punishment number.
- Each PUNISHMENT has a punishment type (withdrawing certain privileges, such as seeing visitors, forced labor, etc.).

## 12. LAW VIOLATORS

- LAW VIOLATORS is a weak entity.
- Each LAW VIOLATOR has the name of the prisoner as a partial key.

## 13. WARDEN

- The PRISON WARDEN is an official in charge of the prison including STAFF and PRISONERS.
- The WARDEN has a warden ID.
- The WARDEN has a composite attribute name (first name, last name).
- The WARDEN has a salary.
- The WARDEN has a gender.
- The WARDEN has an address.
- The WARDEN has a password to the system.

## **14. CANTEEN**

## **15. PRISON YARD**

## **16. HEALTHCARE**

## **17. PRISON COOK**

## **18. HEAD DOCTOR**

## **19. YARD OFFICERS**

## **20. PRISON OFFICERS**

### **1.3 Business Rules**

#### **CELLS:**

- Each prisoner must be [PLACED] in one CELL, each CELL may have many PRISONERS [PLACED] in it (meaning: all prisoners must be placed in cells but a cell can have one or more prisoners).
- A cell is considered fully occupied only if it is being occupied by two PRISONERS otherwise, a new PRISONER will be [PLACED] to the cell by the system.

#### **PRISONERS:**

- Each prisoner must be [PLACED] in one CELL (must be within each cell's capacity).
- The constraints are mentioned in CELLS rules (above).

#### **HEALTH RECORDS:**

- Each PRISONER must [HAS] one HEALTH RECORD, each HEALTH RECORD must belong to one PRISONER.
- Prisoners' HEALTH RECORDS are related to their existence (when a prisoner is released, their health record is automatically deleted).

#### **STAFF:**

- Each PRISONER must be [MANAGED] by many STAFF members, each STAFF member may manage many PRISONERS.
- Each PRISON COOK must [WORKS\_IN] one CANTEEN, each CANTEEN must have one PRISON COOK.

- Each YARD OFFICER must [**OBSERVES**] one PRISON YARD, each PRISON YARD must be observed by many YARD OFFICERS.
- Each HEAD DOCTOR must [**HEADS**] one HEALTHCARE, each HEALTHCARE must have one HEAD DOCTOR.
- HEAD DOCTOR IS-A STAFF
- PRISON OFFICERS IS-A STAFF
- PRISON COOK IS-A STAFF
- YARD OFFICERS IS-A STAFF

## **VISITS:**

- Each PRISONER may [**RECEIVE**] many VISITS, each VISIT must be for one PRISONER.

## **FLOOR:**

- Each FLOOR must be [**SURVEILLANCED**] by many PRISON OFFICERS, each PRISON OFFICER must [**SURVEILLANCED**] one CELLS' FLOOR.
- FLOORS are further grouped into {CELL'S FLOOR, WARDEN'S FLOOR} based on the floor's type.

## **CELL'S FLOOR:**

- Each CELL'S FLOOR must [**CONTAINS**] many CELLS, each CELL must be on one CELL'S FLOOR.

## **WARDEN'S FLOOR:**

- This entity has no relationships.

## **CRIMINAL RECORDS:**

- Each PRISONER must [**OWNS**] one CRIMINAL RECORDS, each CRIMINAL RECORD must belong to one PRISONER.

## **ACTIVITIES:**

- Each ACTIVITY must be [ASSIGNED\_TO] one PRISONER, each PRISONER may have many ACTIVITIES.

## **PUNISHMENT:**

- Each PUNISHMENT must [DISCIPLINES] one LAW VIOLATOR, each LAW VIOLATOR must be disciplined by one punishment.

## **WARDEN:**

- Each WARDEN must [OVERSEES] many STAFF members, each STAFF member must be overseen by one WARDEN.
- Each WARDEN must [DECIDES] many PUNISHMENTS, Each PUNISHMENT must be decided by one warden.

## **1.4 Intended Output of the system**

- Transaction**

- Insert staff members into the system.
- Delete a prisoner from the system.
- Delete a staff member from the system.
- Assign a prisoner to an available cell
- Update cell state
- Assign a prison officer on a specific floor
- Update prisoners' health record
- Update health recorders for prisoners
- Insert a crime in a prisoner's criminal record
- Assign a prisoner as a law violator
- Insert a new visitor's details
- Update a yard officer
- Assign an activity for a prisoner
- Update number of visits
- Assign a new prison cook for the canteen
- ..... etc.

- Queries/output**

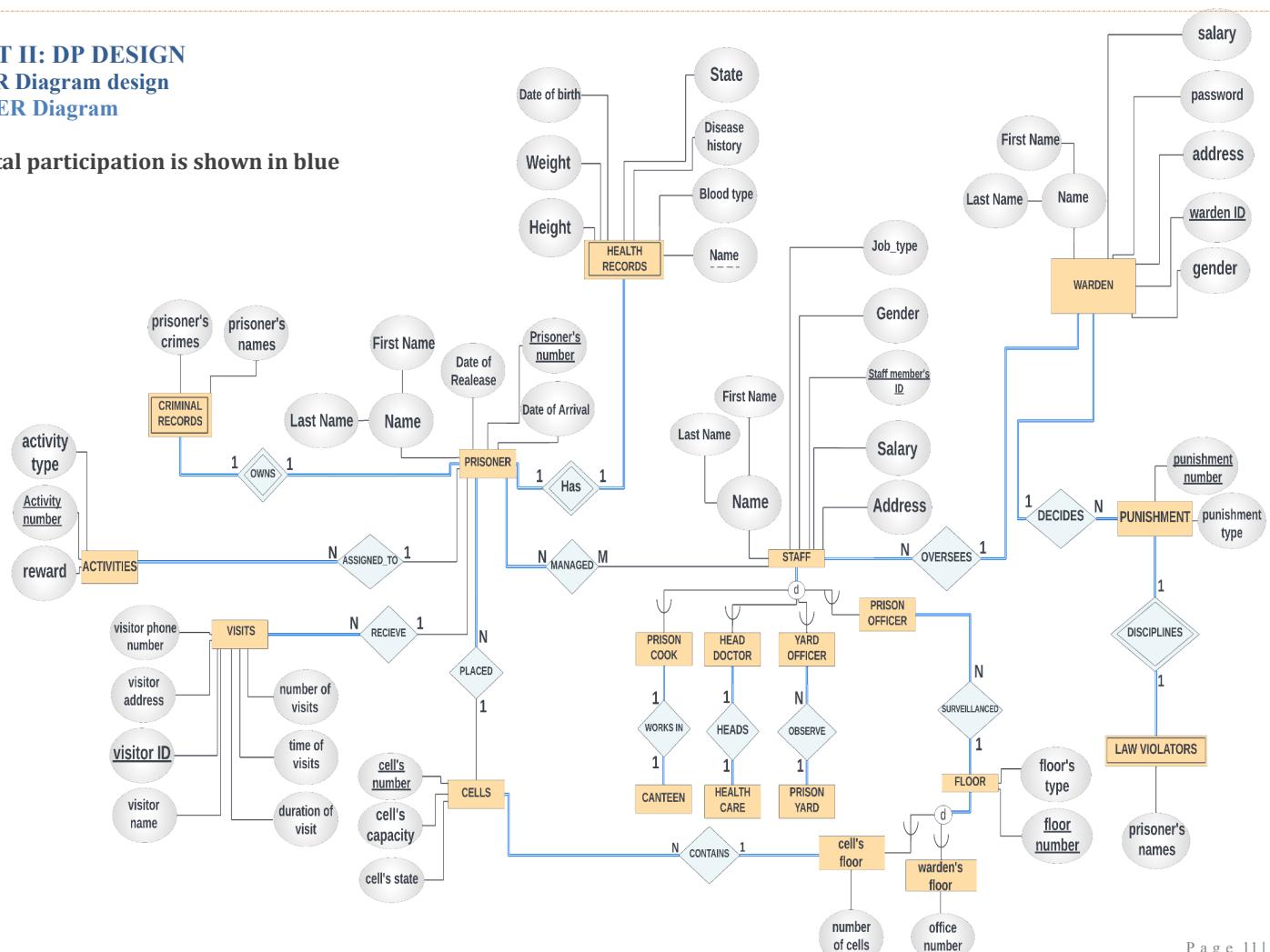
- Search for a cell number by a cell type
- Display visitors of a specific prisoner
- Display prisoners' activity type
- Search criminal records for a prisoner by the prisoner ID
- Display occupied cells
- Display number of unoccupied cells
- Count the remaining years until the release date for a prisoner
- Display the remaining time until visiting hours are over
- Count number of staff-by-staff ID
- Display reward for a prisoner by activity number
- Display all prisoners with their criminal records
- Search address by staff ID
- .....etc.

## PART II: DP DESIGN

### 2 EER Diagram design

#### 2.1 EER Diagram

- Total participation is shown in blue



## 2.2 Design of Business Rules

Business Rules	Design Decisions	Justifications (if any)
Many PRISONERS must be [PLACED] in one CELL.	N:1 binary relationship between PRISONERS and CELL.	<p>Each PRISONER must be [PLACED] in one CELL, each CELL may have many PRISONERS [PLACED] in it.</p> <p>It is total participation on the PRISONERS' side since all PRISONERS will be placed in CELLS, and it is partial participation on the CELLS' side since CELLS can be occupied by one or more PRISONERS.</p>
Each PRISONER must [HAS] one HEALTH RECORD that is related to them as a weak entity.	1:1 binary identifying relationship between PRISONER and HEALTH RECORDS.	<p>Each PRISONER must [HAS] one HEALTH RECORD, each HEALTH RECORD must belong to one PRISONER</p> <p>If a PRISONER does not exist, their HEALTH RECORD is unnecessary, since its existence depends on a PRISONER, it is shown in total participation on both sides.</p>
STAFF members must have a job type.	Superclass/subclass relationship with disjointness and completeness constraints.	<p>Total participation since each STAFF member must have a job type.</p> <p>Each STAFF member must have one job type; therefore, it is disjointed.</p>
many PRISONERS must be [MANAGED] by many prison STAFF.	N: M binary relationship between PRISONERS and STAFF	<p>Each PRISONER must be [MANAGED] by many STAFF members, each STAFF member may manage many PRISONERS.</p> <p>It is total participation on the PRISONERS' side because PRISONERS will be managed only by STAFF members. On the other hand, it is partial participation since STAFF has different jobs.</p>

Each PRISON COOK must [WORKS_IN] in one CANTEEN.	1:1 binary relationship between PRISON COOK and CANTEEN.	<p>Each PRISON COOK must [WORKS_IN] one CANTEEN, each CANTEEN must have one PRISON COOK</p> <p>It is total participation on both sides since PRISON COOK must work only in the CANTEEN and the CANTEEN must have only one PRISON COOK.</p>
Many YARD OFFICERS [OBSERVE] one PRISON YARD.	N:1 binary relationship between YARD OFFICERS and PRISON YARD.	<p>Each YARD OFFICER must [OBSERVES] one PRISON YARD, each PRISON YARD must be observed by many YARD OFFICERS.</p> <p>Many YARD OFFICERS must observe one PRISON YARD, and the only PRISON YARD must be observed by many YARD OFFICERS, hence, it is total participation on both sides.</p>
One HEAD DOCTOR must [HEADS] the prisons HEALTHCARE	1:1 binary relationship between HEAD DOCTORS and HEALTHCARE	<p>Each HEAD DOCTOR must [HEADS] one HEALTHCARE, each HEALTHCARE.</p> <p>It is total participation on both sides since the only HEAD DOCTOR will head the only healthcare, and the healthcare will only have one head doctor.</p>
Each PRISONER can [RECEIVE] a certain number of VISITS a month.	1: N binary relationship between PRISONER and VISIT.	<p>Each PRISONER may [RECEIVE] many VISITS, each VISIT must be for one PRISONER.</p> <p>It is partial participation on the PRISONERS' side because not all PRISONERS will receive VISITS and it is total participation on the VISITS' side because many VISITS can be received by one PRISONER.</p>

FLOOR must have a floor type.	Superclass/subclass relationship with disjointness and completeness constraints.	Total participation since every FLOOR has a floor type.  Each FLOOR has one floor type; therefore, it is disjointed.
Each FLOOR must be [SURVEILLANCED] by many PRISON OFFICERS.	1: N binary relationship between FLOOR and PRISON OFFICERS.	Each FLOOR must be [SURVEILLANCED] by many PRISON OFFICERS, each PRISON OFFICER must [SURVEILLANCED] one CELLS' FLOOR.  All FLOORS will be surveillanced by all PRISON OFFICERS and all PRISON OFFICERS will surveillance all FLOORS. Hence, it is total participation on both sides.
Each CELL'S FLOOR must [CONTAINS] many CELLS.	1: N binary relationship between CELL'S FLOOR and CELLS.	Each CELL'S FLOOR must [CONTAINS] many CELLS, each CELL must be on one CELL'S FLOOR.  It is total participation on both sides because all CELL'S FLOORS will contain CELLS and all CELLS will be on CELL'S FLOOR.
Each PRISONER must [OWNS] one CRIMINAL RECORDS.	1:1 binary relationship between PRISONERS and CRIMINAL RECORDS.	Each PRISONER must [OWNS] one CRIMINAL RECORD and each CRIMINAL RECORD belongs to only one PRISONER.  It is total participation on both sides because all prisoners must own a CRIMINAL RECORD and all CRIMINAL RECORDS belong to PRISONERS.

<p>Many ACTIVITIES may be [ASSIGNED_TO] each PRISONER.</p>	<p>N: 1 relationship between ACTIVITIES and PRISONERS.</p>	<p>Each ACTIVITY must be [ASSIGNED_TO] one PRISONER, each PRISONER may have many ACTIVITIES.</p>
<p>One PUNISHMENT must [DISCIPLINES] each LAW VIOLATOR LAW VIOLATOR IS related to PUNISHMENT as a weak entity.</p>	<p>1:1 binary identifying relationship between PUNISHMENT and LAW VIOLATORS.</p>	<p>Each PUNISHMENT must [DISCIPLINES] one LAW VIOLATOR, each LAW VIOLATOR must be disciplined by one punishment.  If a LAW VIOLATOR does not exist, the PUNISHMENT is unnecessary, because its existence depends on the LAW VIOLATOR, it is shown in total participation.</p>
<p>One prison WARDEN must [OVERSEES] all STAFF members.</p>	<p>1: N binary relationship between WARDEN and STAFF.</p>	<p>Each WARDEN must [OVERSEES] many STAFF members, each STAFF member must be overseen by one WARDEN.  Total participation on both sides because the only WARDEN will oversee all STAFF members and all STAFF members will be overseen by the only WARDEN</p>
<p>WARDEN must [DECIDES] all PUNISHMENTS.</p>	<p>N:1 binary relationship between PUNISHMENT and WARDEN.</p>	<p>Each WARDEN must [DECIDES] many PUNISHMENTS, Each PUNISHMENT must be decided by one warden.  Total participation on both sides because the WARDEN will decide all the PUNISHMENTS, and all PUNISHMENTS will only be decided by the WARDEN.</p>

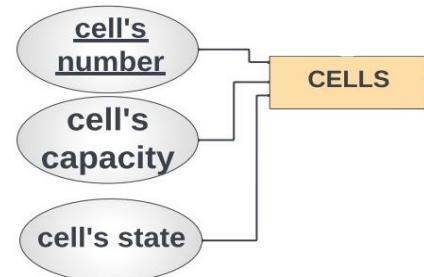
### 3 ER-to-logical schema mapping

#### 3.1 Mapping of Regular Entity Types

- For each regular entity create a new relation and include all the simple attributes, then choose one of the key attributes as the primary key.

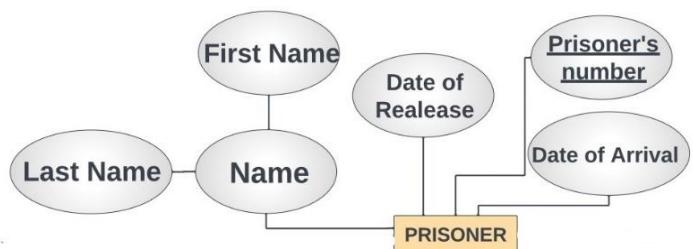
CELLS

<u>Cell's number</u>	Cell's capacity	Cell's state



PRISONERS

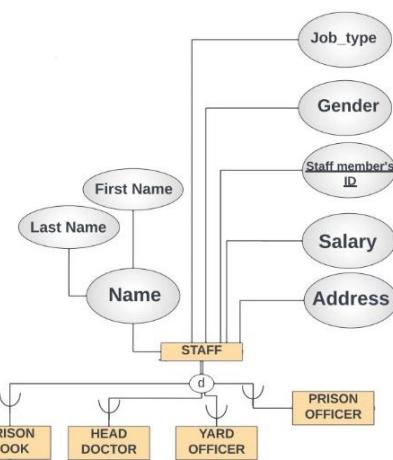
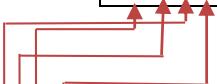
<u>Prisoner number</u>	First name	Last name	Date of arrival	Date of release



- We used the general way to map the STAFF specialization (EER) since it will benefit us in the implementation phase.

## STAFF

<u>Staff member's ID</u>	First name	Last name	Gender	Job type	Salary	Address
--------------------------	------------	-----------	--------	----------	--------	---------



## PRISON COOK

<u>ID</u>
-----------

## HEAD DOCTOR

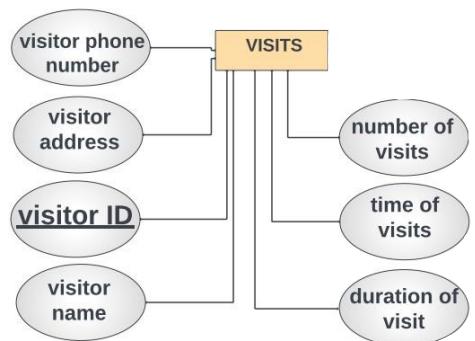
<u>ID</u>
-----------

## YARD OFFICERS

<u>ID</u>
-----------

## PRISON OFFICERS

<u>ID</u>
-----------

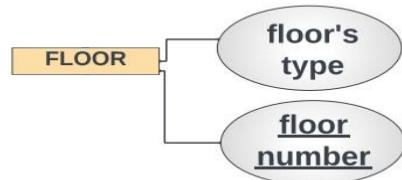


## VISITS

<u>Visitor ID</u>	Visitor Name	Visitor Phone number	Visitor Address	Duration Of visit	Time of visits	Number of visits

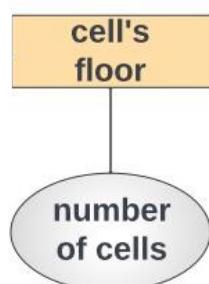
## FLOOR

<u>Floor number</u>	Floor's type



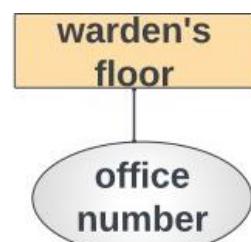
## CELL'S FLOOR

<u>F number</u>	Number of cells



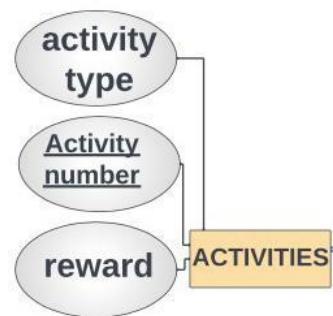
## WARDEN'S FLOOR

<u>F number</u>	office number



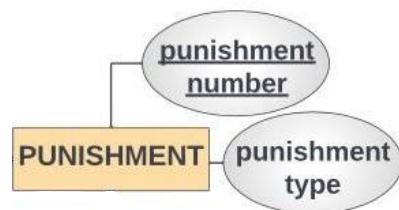
## ACTIVITIES

<u>Activity number</u>	Reward	Activity type



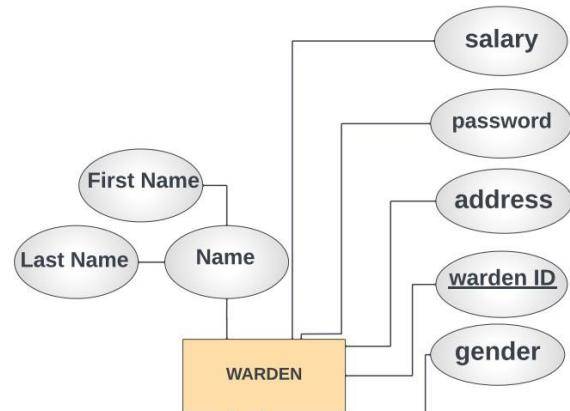
## PUNISHMENT

<u>Punishment number</u>	Punishment type



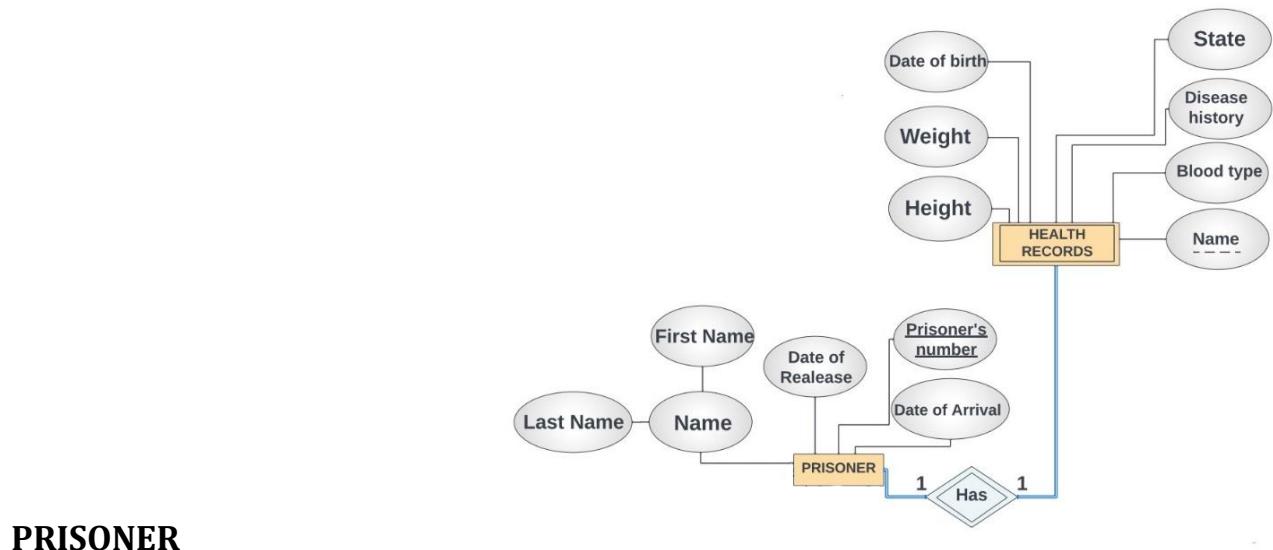
## WARDEN

<u>Warden ID</u>	First name	Last name	Gender	Address	Password	Salary



### 3.2 Mapping of Weak Entity Types

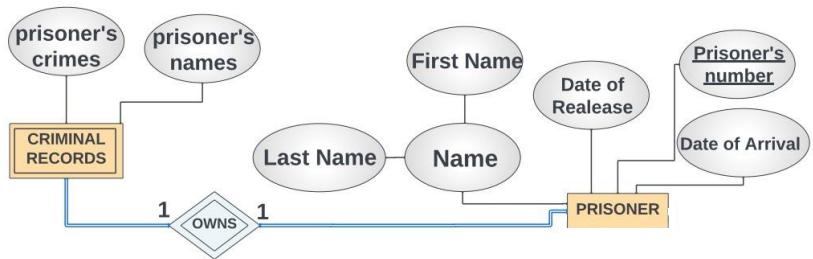
- Include in the weak entity the primary key of the identifying entity as a foreign key and the primary key of the weak entity are the combinations of the primary key of the identifying entity and the partial key of the weak entity.



<u>Prisoner number</u>	First name	Last name	Date of arrival	Date of release
----------------------------	------------	-----------	--------------------	-----------------

HEALTH RECORDS

<u>Prisoner's name</u>	<u>P_number</u>	Blood type	State	Date of birth	Weight	Height	Disease history
----------------------------	-----------------	---------------	-------	------------------	--------	--------	--------------------



## PRISONERS

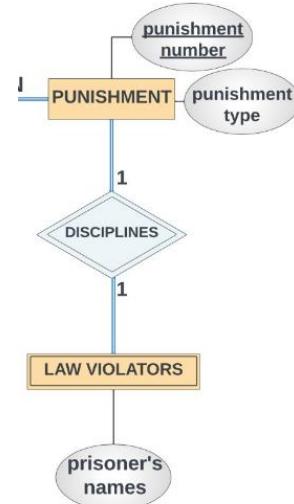
<u>Prisoner number</u>	First name	Last name	Date of arrival	Date of release

## CRIMINAL RECORDS

<u>Prisoner's name</u>	<u>P number</u>	Prisoner's crime

## PUNISHMENT

<u>Punishment number</u>	Punishment type



## LAW VIOLATORS

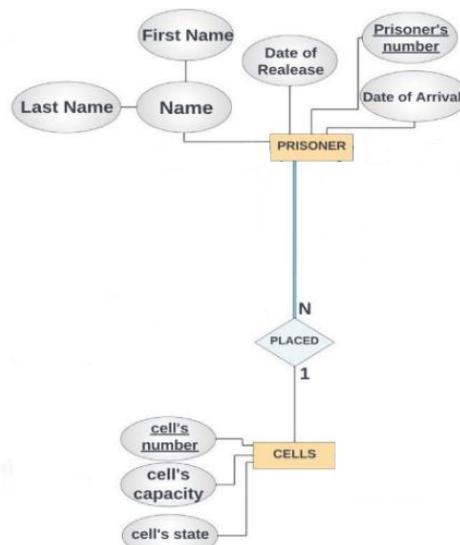
<u>Prisoner's name</u>	<u>Punish number</u>

### 3.3 MAPPING OF BINARY 1:1 RELATIONSHIP TYPES

- Shown in (Mapping of Weak Entity Types and Regular Entity Types).

### 3.4 MAPPING OF 1:N RELATIONSHIP TYPES

- To map the binary 1:N relationship, we include a foreign key on the N side.

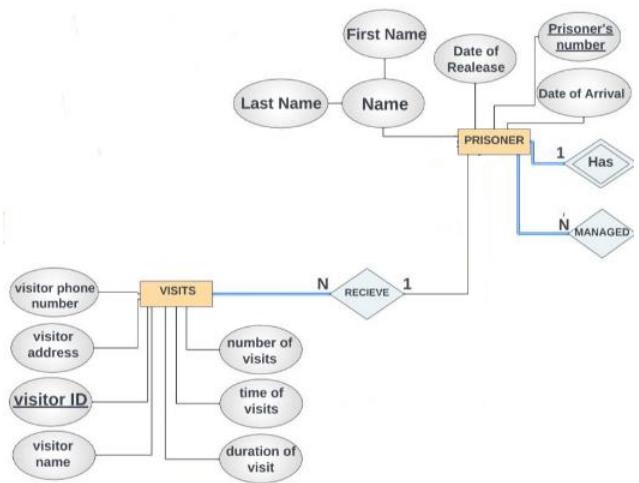


#### CELLS

<u>Cell number</u>	Cells state	Cell capacity

PRISONERS

<u>Prisoner number</u>	C_number	First name	Last name	Date of arrival	Date of release

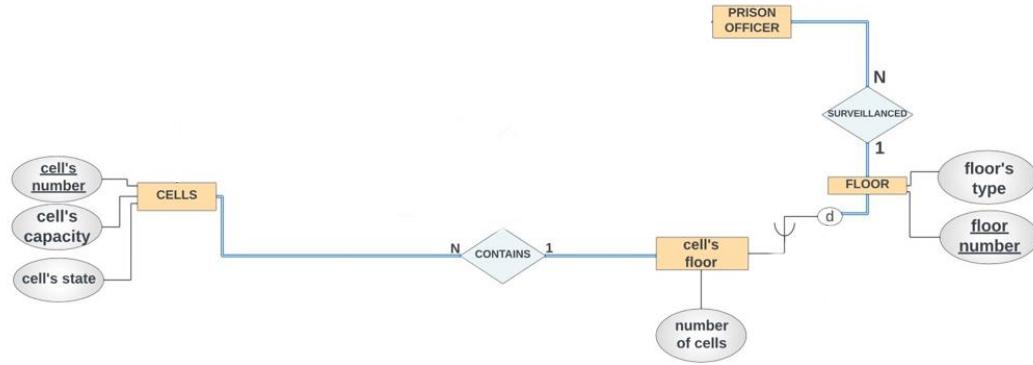


## PRISONERS

<u>Prisoner number</u>	First name	Last name	Date of arrival	Date of release

## VISITS

<u>Visitor ID</u>	P_number	Visitor Address	Phone Number	Visitor Name	Duration Of visit	Time of visit	Number of visits



## Cell's floor

<u>Floor number</u>	Number of Cells

## Cells

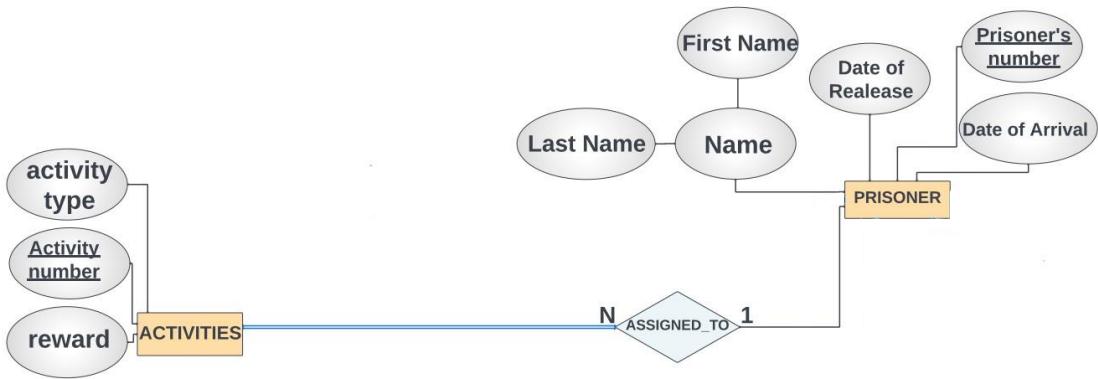
<u>Cell number</u>	F_number	Cells state	Cell capacity

## FLOOR

<u>Floor number</u>	Number of cells	Prison officer ID

## PRISON OFFICERS

<u>ID</u>	F_number



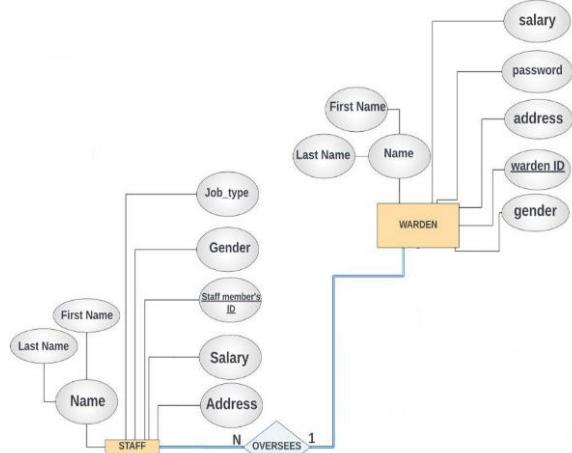
## PRISONERS

<u>Prisoner number</u>	First name	Last name	Date of arrival	Date of release

## ACTIVITIES

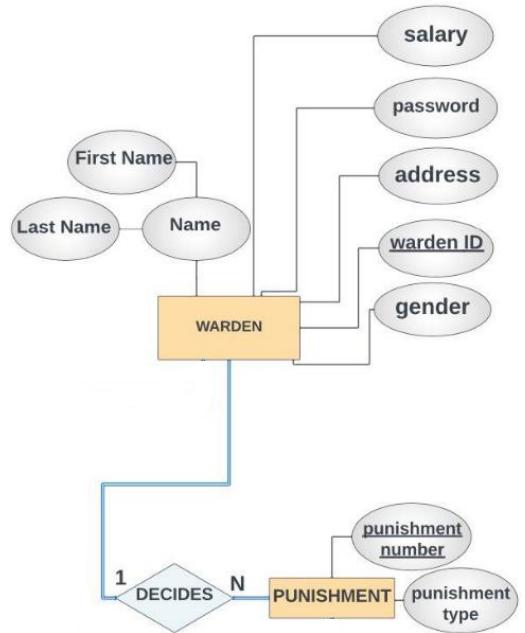
Activity number	P_number	Reward	Activity type

## WARDEN



## STAFF

<u>Staff member's ID</u>	W_ID	First name	Last name	Gender	Job type	Salary	Address



## WARDEN

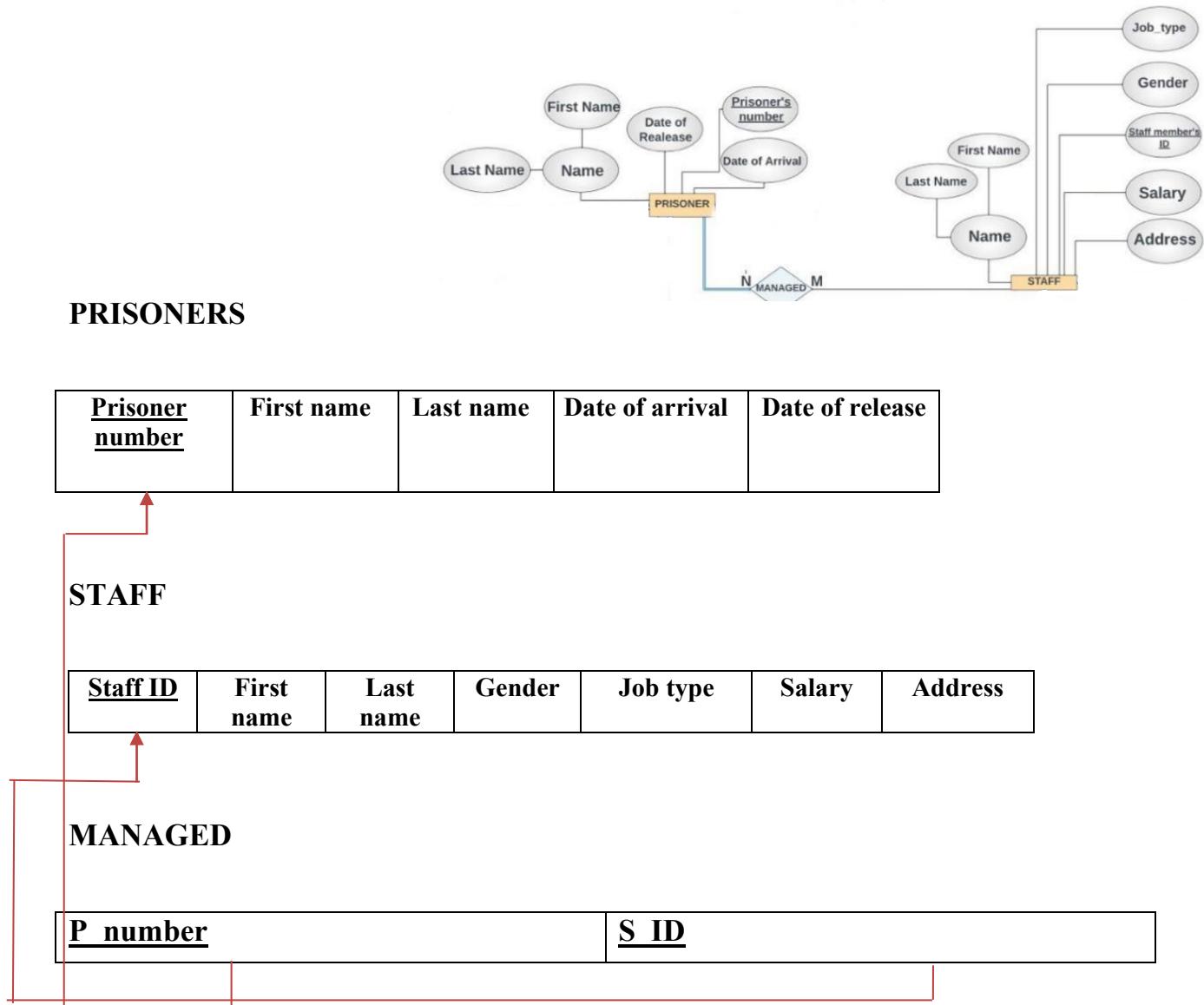
<u>Warden ID</u>	First name	Last name	Gender	Address	Password	Salary
------------------	------------	-----------	--------	---------	----------	--------

## PUNISHMENT

<u>Punishment number</u>	<u>W_ID</u>	Punishment type
--------------------------	-------------	-----------------

### 3.5 MAPPING OF M:N RELATIONSHIP TYPES

- Create a new relation for the relationship then combine the primary keys of both entities to form the primary key of the new relation.



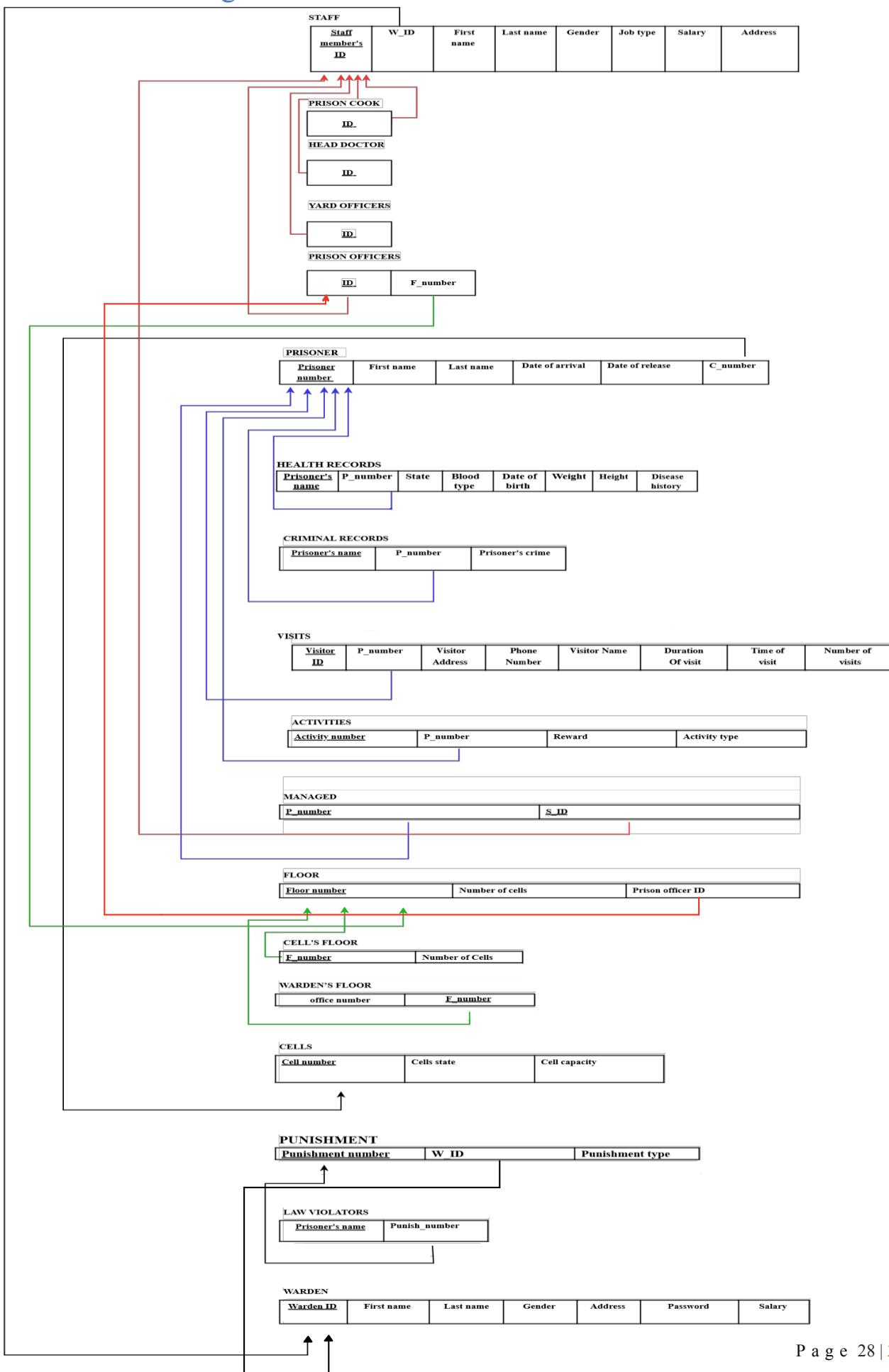
### 3.6 Mapping of multivalued attributes:

- None

### 3.7 Mapping of n-ary relationship types:

- None

### 3.8 Schema diagram



## 4 Normalization

### 4.1 First Normal Form

In the First normal form (1NF), composite, multivalued attributes, and nested relations are not allowed. Hence, we only include attributes with atomic values in their domains. Our schema has neither nested relation nor multivalued attributes. We only have one composite attribute which is (name) each sub value was given in a column. For that reason, we can conclude that our relations do not violate any guidelines for the first normal form.

STAFF							
Staff member's ID	W_ID	First name	Last name	Gender	Job type	Salary	Address

PRISON COOK

ID
----

HEAD DOCTOR

ID
----

YARD OFFICERS

ID
----

PRISON OFFICES

ID	F_number
----	----------

PRISONERS

Prisoner number	C_number	First name	Last name	Date of arrival	Date of release
-----------------	----------	------------	-----------	-----------------	-----------------

HEALTH RECORDS

Prisoner's name	P_number	State	Blood type	Date of birth	Weight	Height	Disease history
-----------------	----------	-------	------------	---------------	--------	--------	-----------------

CRIMINAL RECORDS

Prisoner's name	P_number	Prisoner's crime
-----------------	----------	------------------

VISITS

Visitor ID	P_number	Visitor Name	Visitor Phone number	Visitor Address	Duration Of visit	Time of visits	Number of visits
------------	----------	--------------	----------------------	-----------------	-------------------	----------------	------------------

ACTIVITIES

Activity number	Reward	Activity type	P_number
-----------------	--------	---------------	----------

MANAGED

P_number	S_ID
----------	------

FLOOR

Floor number	Number of cells	Prison officer ID
--------------	-----------------	-------------------

CELL'S FLOOR

F_number	Number of cells
----------	-----------------

WARDENS FLOOR

Office number	F_number
---------------	----------

CELLS

Cell's number	Cell's capacity	Cell's state
---------------	-----------------	--------------

PUNISHMENT

Punishment number	W_ID	Punishment type
-------------------	------	-----------------

LAW VIOLATORS

Prisoner's name	Punish number
-----------------	---------------

WARDEN

Warden ID	First name	Last name	Gender	Address	Password	Salary
-----------	------------	-----------	--------	---------	----------	--------

## 4.2 Second Normal Form

To make our relational schema in Second normal form [2NF], every non-prime attribute should be fully functionally dependent on the primary key. Full FD: means that the removal of the primary key attribute (or an attribute that is part of a PK) results in losing the functional dependency.

If the relation has a single attribute only, then it will be in a Second Normal Form automatically.

### STAFF

<u>Staff member's Id</u>	W_ID	First name	Last name	Gender	Job type	Salary	Address

{Staff member's ID}  $\rightarrow$  W\_ID, Fname, Lname, gender, job type, salary and address are all functionally dependent [FD] on the Primary key, and is it a full FD.

### PRISONERS

<u>Prisoner number</u>	C_number	First name	Last name	Date of arrival	Date of release

{Prisoner Number}  $\rightarrow$  C\_number, first name, last name, date of arrival and date of release are full FD.

### HEALTH RECORDS

<u>Prisoner's name</u>	P_number	State	Blood type	Date of birth	Weight	Height	Disease history

{Prisoner's name and P\_number }  $\rightarrow$  State, blood type, date of birth, weight, height and disease history are full FD.

### VISITS

<u>Visitor ID</u>	P_number	Visitor Name	Visitor Phone number	Visitor Address	Duration Of visit	Time of visits	Number of visits

{Visitor ID}  $\rightarrow$  P\_number, Visitor Name, Visitor Phone number, Visitor Address, Duration of visit, Time of visits and Number of visits are full FD.

## ACTIVITIES

<u>Activity number</u>	Reward	Activity type	P_number

{Activity number} -> Reward, and Activity type and P\_number are full FD.

## CELLS

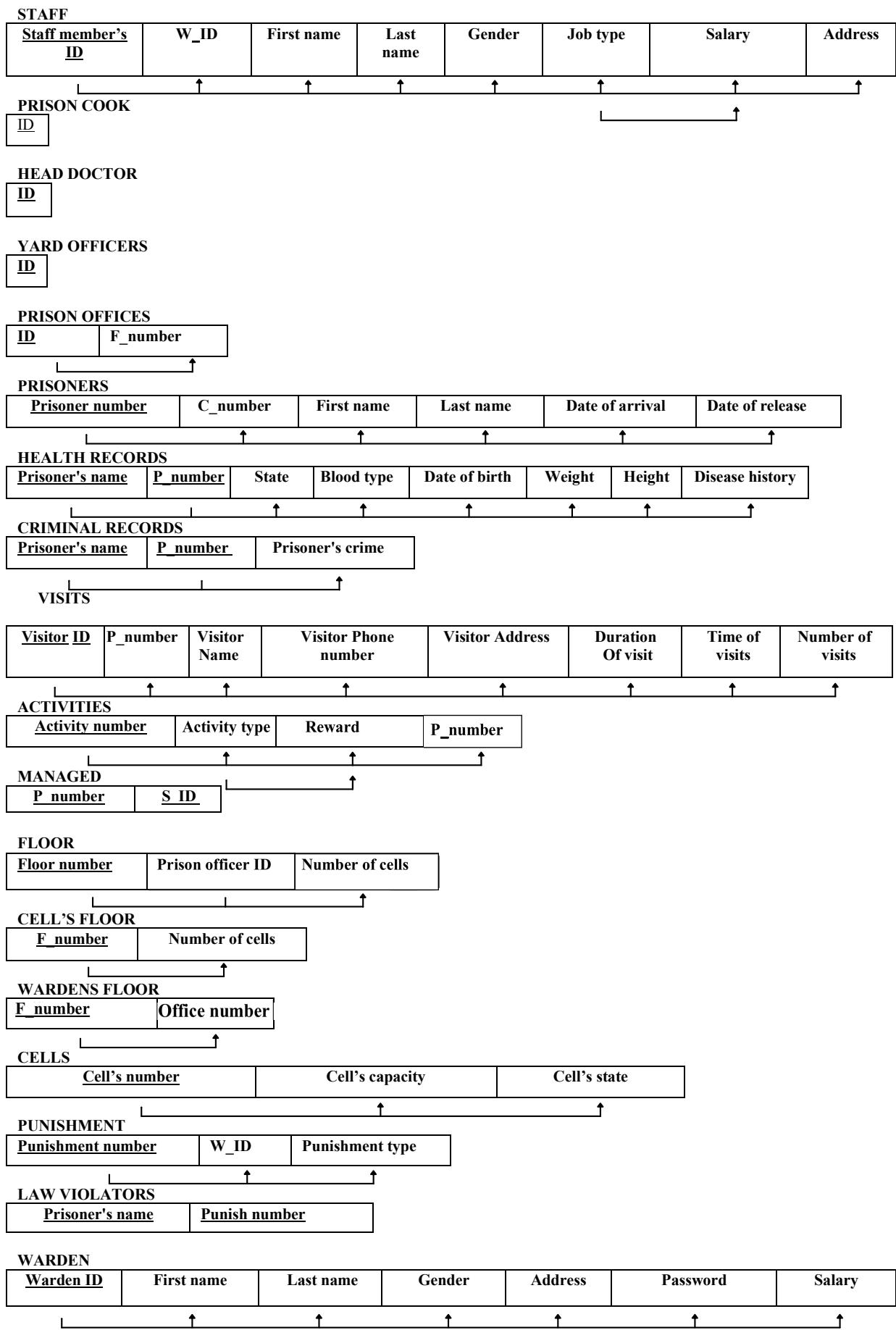
<u>Cell's number</u>	Cell's capacity	Cell's state

{Cell's number} -> Cell's capacity and Cell's state are full FD.

## WARDEN

<u>Warden ID</u>	First name	Last name	Gender	Address	Password	Salary

{Warden ID}-> Fname, Lname, Gender, Address, Password, and Salary are full FD.



### 4.3 Third Normal Form

-To make the relational schema in Third normal form [3NF], there should not be a Transitive dependency. Whenever some indirect relationship happens to cause functional dependency (FD), it is known as Transitive Dependency. Thus, if  $A \rightarrow B$  and  $B \rightarrow C$  are true, then  $A \rightarrow C$  happens to be a transitive dependency. Therefore, STAFF relation violates the 3NF guideline as explained below.

- Staff members ID  $\rightarrow$  Job type and Job type  $\rightarrow$  Salary.
- Since Job type is a non-prime attribute, and Salary is a non-prime attribute, hence the relation STAFF is not in 3NF.
- To normalize it, we decompose STAFF into the two 3NF relations:

#### STAFF

<u>Staff member's Id</u>	W_ID	First name	Last name	Gender	Job type	Salary	Address

#### JOB\_SALARY

<u>Job type</u>	Salary

#### STAFF

<u>Staff member's ID</u>	W_ID	First name	Last name	Gender	Address	Job type

- Activity number  $\rightarrow$  Activity type and Activity type  $\rightarrow$  Reward
- Since Activity type is a non-prime attribute, and Reward is a non-prime attribute, hence the relation ACTIVITIES is not in 3NF.
- To normalize it, we decompose ACTIVITIES into the two 3NF relations:

#### ACTIVITIES

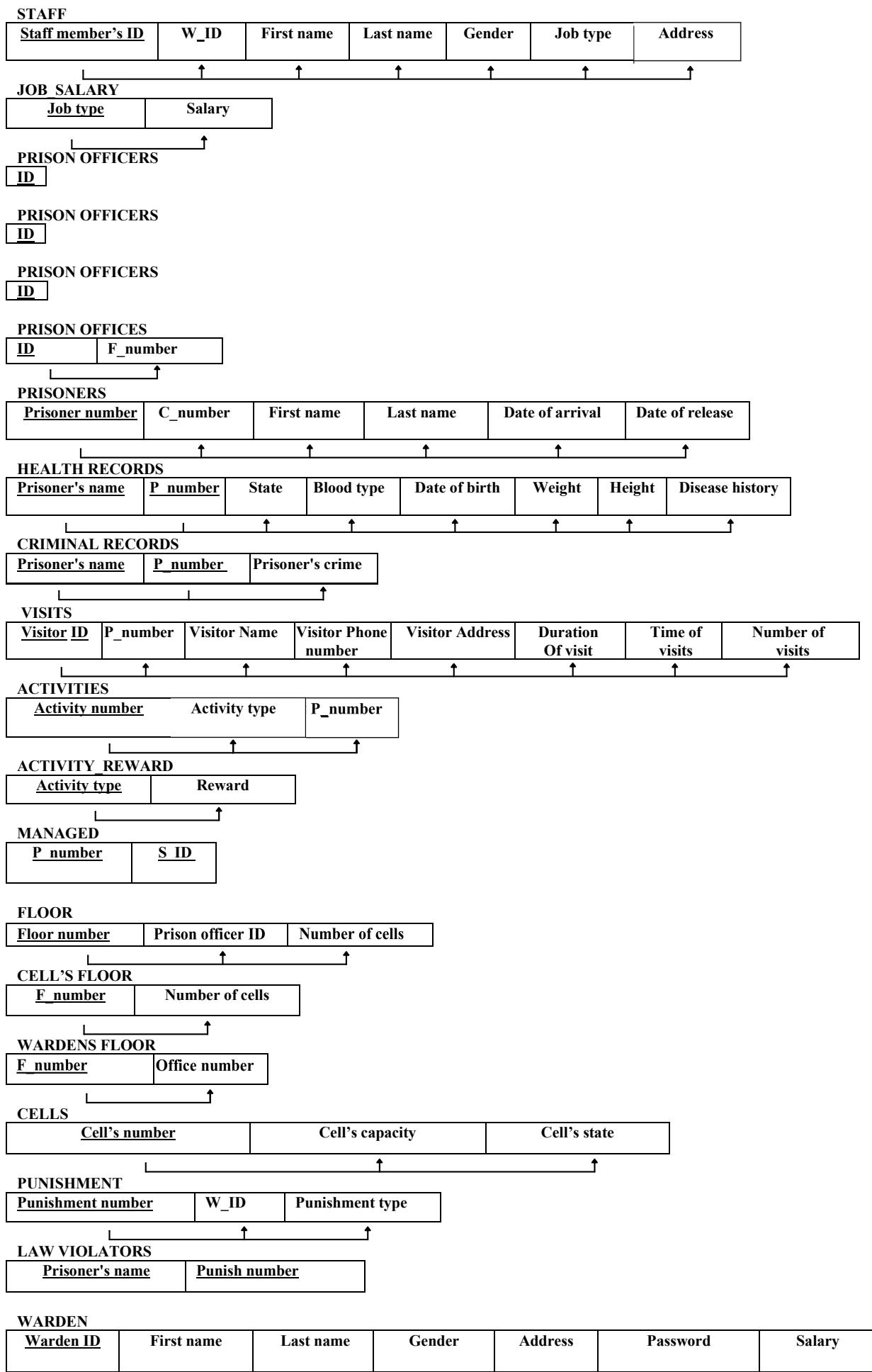
<u>Activity number</u>	Reward	Activity type	P_number

#### ACTIVITY\_REWARD

<u>Activity type</u>	Reward

#### ACTIVITIES

<u>Activity number</u>	Activity type	P_number



## 5 Final DB Schema Diagram

