“Hospital management system” Part 1

BS18-04

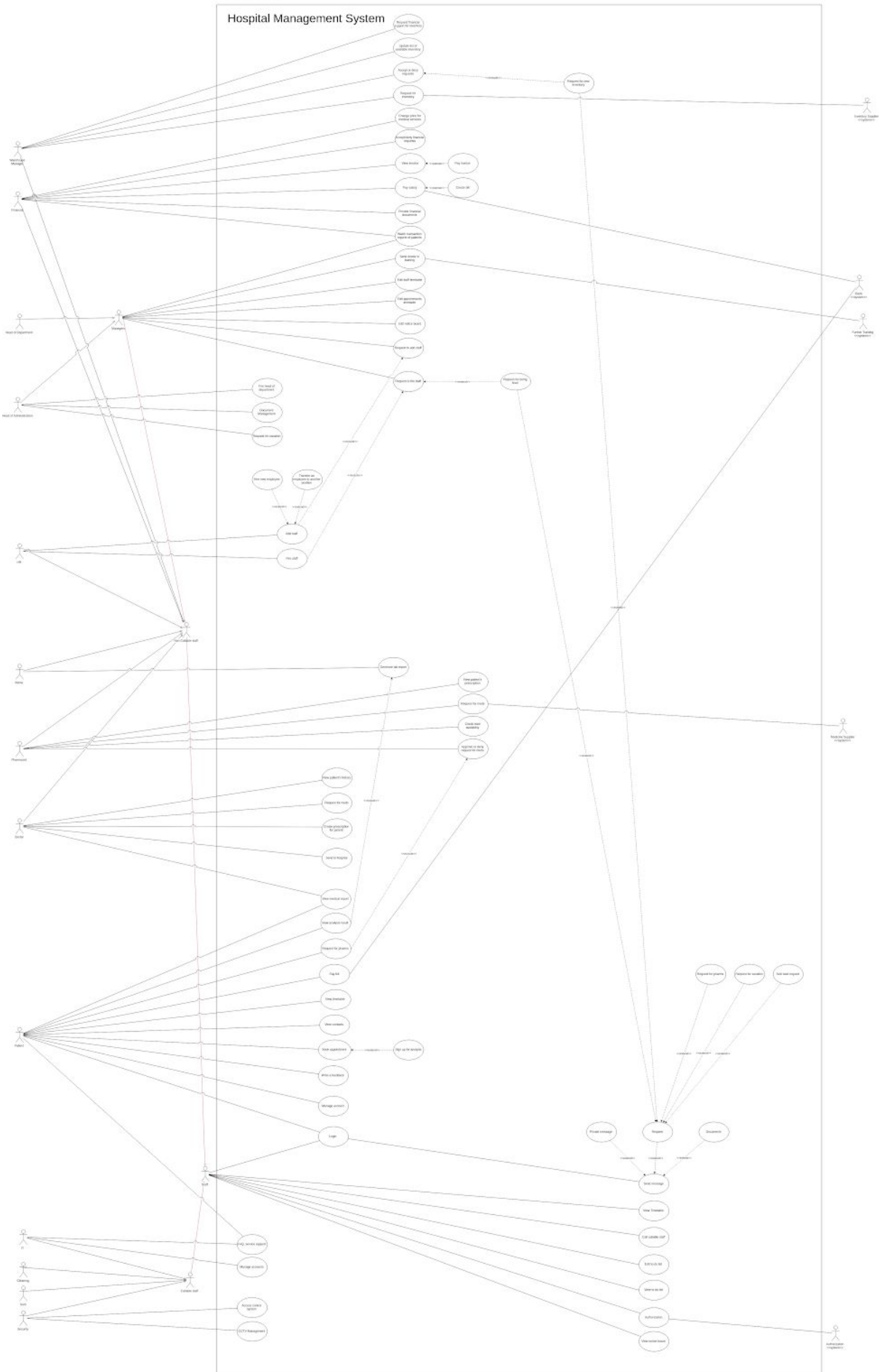
Askarbek Kazybek, Bogdanova Alina, Dubina Nikita, Smirnov Nikita, Zarubin Yurii

Description

The “Hospital management system” contains three subsystems, connected to the staff, administration and treatment. Together they form one connected system, the blocks described above are used only to simplify the use-case diagram, make it more readable. We removed the reception staff in the hospital and our system does not have a certain number of departments, therefore it is expandable and is suitable for different types of medical institutions, including some specific ones (gynaecology, maternity hospital, mental hospital, etc.). We chose this implementation because we could ask our relatives, who works in the hospital, and receive more information about the system.

Use-case-diagram

Askarbek Kazybek
Bogdanova Alina
Nikita Dubina
Nikita Smirnov
Zarubin Yurii



Functional requirements

1. General

Requirement ID	1.1
Title	System should allow any user to login
Type	Functional
Description	Patient should be able to login into account using their login and password
Priority	1
Risk	C

1. Patient account

Requirement ID	1.1
Title	System should allow Authorized patient user to view medical report feature for patient account
Type	Functional
Description	Authorized patient should be able to see a medical report for each attend respectively
Priority	1
Risk	C

Requirement ID	1.2
Title	System should allow Authorized patient user to manage account
Type	Functional
Description	Authorized patient should be able to manage account, see and change its login and password
Priority	1
Risk	H

Requirement ID	1.3
Title	System should allow Authorized patient user to pay bills
Type	Functional
Description	Authorized patient should be able to pay the bill that was given by the financial department of the hospital
Priority	1
Risk	M

Requirement ID	1.4
Title	System should allow Patient user to view appointments timetable
Type	Functional
Description	Authorized patient should be able to see the timetable of appointments with doctors
Priority	1
Risk	L

Requirement ID	1.5
Title	System should allow Authorized patient user to book appointment
Type	Functional
Description	Authorized patient should be able to book their appointment, choose date and time, doctor, select cause of appointment
Priority	1
Risk	L

Requirement ID	1.6
Title	System should be able to provide a feedback feature for Authorized patient user
Type	Functional
Description	Authorized patient should be able to write feedback as on hospital in general, as on some specific doctor. Both types of feedback are submitted and shown together. One, who send feedback should have a choice to submit feedback anonymously or not.
Priority	2
Risk	L

Requirement ID	1.7
Title	System should allow Authorized patient user to get the results of analysis
Type	Functional
Description	Authorized patient should be able to view the results of analysis done in the hospital
Priority	2
Risk	L

Requirement ID	1.8
Title	System should allow any user to get contacts
Type	Functional
Description	Any user (even one who is neither authorized nor registered) should be able to view information about the location of the hospital, its contacts: phones of the ambulance and reception and FAQ
Priority	2
Risk	L

Requirement ID	1.9
Title	System should allow any user to view feedback
Type	Functional
Description	Any user (even one who is neither authorized nor registered) should be able to view the feedback on doctors and hospital
Priority	3
Risk	L

Requirement ID	1.10
Title	System should be able to provide “request for pharms feature” for authorized patient user
Type	Functional
Description	Authorized patient should be able to request for pharms
Priority	3
Risk	L

2. Staff

Requirement ID	2.1
Title	System should allow Staff members to write message
Type	Functional
Description	Staff members should be able to write a message to the administration, asking salary, vacation or to solve issues. Message can contain text, image or file (max summary size is 10 MB).
Priority	2
Risk	M

Requirement ID	2.2
Title	System should allow Staff member user to view timetable
Type	Functional
Description	Staff member should be able to see the staff's timetable, when they could make an appointment and use patient’s room
Priority	1
Risk	L

Requirement ID	2.3
Title	System should be able to support TODO List and provide its control to Staff member user
Type	Functional
Description	Staff members should be able to manage TODO list add new tasks into it, and remove them. Each staff member should have their own TODO list.
Priority	2
Risk	L

Requirement ID	2.4
Title	System should allow Staff member user to view notice board
Type	Functional
Description	Staff should be able to see notice board: check for events and holidays
Priority	3
Risk	L

3. Manager

Requirement ID	3.1
Title	System should allow Manager user to edit hospital's timetables
Type	Functional
Description	Manager should be able to edit the main timetables of hospital: hospital staff's timetable and timetable of appointments, which is accessible to patients
Priority	1
Risk	C

Requirement ID	3.2
Title	System should allow Manager user to add and fire staff
Type	Functional
Description	Manager should be able to request HR to search for new stuff and to prepare documents for firing. Only the head of administration is able to fire or add head of department.
Priority	2
Risk	M

Requirement ID	3.3
Title	System should give an ability to Manager user to send doctor to training
Type	Functional
Description	Manager should be able to send doctors for an additional training: such decision should affect doctors and HRs (to make the docs prepared in a time)
Priority	3
Risk	L

Requirement ID	3.4
Title	System should give an ability to Manager user to edit notice board
Type	Functional
Description	Manager should be able to edit notice board, make notifications about holidays or other events
Priority	3
Risk	L

4. Doctor

Requirement ID	4.1
Title	System should allow Doctor user to send patients to hospital
Type	Functional
Description	Doctors should be able to send patients to hospital and provide reasons.
Priority	2
Risk	H

Requirement ID	4.2
Title	System should allow Doctor user to create prescription to patient
Type	Functional
Description	Doctors should be able to upload medical prescription to patient including diagnosis, this prescription will be available at patient's account
Priority	2
Risk	H

Requirement ID	4.3
Title	System should allow Doctor user to send request for medicals to pharmacist
Type	Functional
Description	Doctor should be able to request medicals for treatment from pharmacist. This request can be approved only by pharmacist
Priority	2
Risk	M

5. Nurse

Requirement ID	5.1
Title	System should allow Nurse user to create and fill lab reports
Type	Functional
Description	Nurses should be able to create and manage lab reports, which will be automatically sent to the patients
Priority	1
Risk	M

6. Security

Requirement ID	6.1
Title	System should be able to support CCTV Management and provide its control to Security user
Type	Functional
Description	Security should be able to - access CCTV system, export of video files - access to the CCTV system status
Priority	2
Risk	M

Requirement ID	6.2
Title	System should allow Security user to have access to the control system
Type	Functional
Description	Security should be able to - control electronic locks, room lighting - have access to the security system status: 1. System of electrical passes 2. Lighting of the hospital
Priority	2
Risk	M

7. IT specialist

Requirement ID	7.1
Title	System should allow IT specialist user to manage accounts
Type	Functional
Description	IT specialist should be able to have full access to the system data except GDPR's constraints: - create, change and remove stuff's accounts - view the general statistics about users and manage user account in case of such a request
Priority	1
Risk	H

Requirement ID	7.2
Title	System should allow IT specialist user to answer another user's question
Type	Functional
Description	IT specialist should be able to answer the question asked by the user (message that was sent by user)
Priority	3
Risk	L

8. Financial

Requirement ID	8.1
Title	System should be able to support feature "Pay invoices"
Type	Functional
Description	Financial manager should be able to pay invoices for the services
Priority	1
Risk	M

Requirement ID	8.2
Title	System should allow financial user to create bill
Type	Functional
Description	Financial manager should be able to request patients to pay for their appointment
Priority	1
Risk	M

Requirement ID	8.3
Title	System should be able to support feature “Pay salary”
Type	Functional
Description	Financial manager should be able to pay salary to the hospital workers
Priority	1
Risk	M

Requirement ID	8.4
Title	System should allow Financial to accept/deny financial requests
Type	Functional
Description	Financial manager should be able to accept or deny financial requests that were made by hospital workers
Priority	2
Risk	M

Requirement ID	8.5
Title	System should allow Financial user to watch transaction reports of patients
Type	Functional
Description	Financial manager should be able to see the history of patient's payments and their transactions
Priority	2
Risk	L

Requirement ID	8.6
Title	System should allow Financial to change price for medical services
Type	Functional
Description	Financial manager should be able to change prices for medical services due to the current financial situation
Priority	2
Risk	L

Requirement ID	8.7
Title	System should allow Financial to provide financial documents
Type	Functional
Description	Financial manager should be able to provide financial documents to the head of department
Priority	2
Risk	L

9. HR

Requirement ID	9.1
Title	System should allow HR to add new stuff
Type	Functional
Description	HR should be able to add information and documents of new staff or prepare documents for changing the position of current employee, ability to make changes about staff in database. Information that should be stored about a staff: - Name - Surname - Job title - Phone number - Salary
Priority	1
Risk	H

Requirement ID	9.2
Title	System should allow HR to fire stuff
Type	Functional
Description	HR should be able to change information about staff in database (delete employees from current staff), prepare documents for firing staff. Information that should be deleted: - Name - Surname - Job title - Phone number - Salary
Priority	1
Risk	H

10. Pharmacist

Requirement ID	10.1
Title	System should allow Pharmacist to approve and deny requests for meds
Type	Functional
Description	Pharmacist should be able to approve or deny requests for medicines from hospital workers. Information that should be stored: - Medicines list - Stuff accounts
Priority	1
Risk	M

Requirement ID	10.2
Title	System should allow Pharmacist to check medicines for availability
Type	Functional
Description	Pharmacist should be able to check medicines for availability, see its quantity. Information that should be stored: - Medicines list
Priority	1
Risk	M

Requirement ID	10.3
Title	System should allow Pharmacist to request for pharms
Type	Functional
Description	Pharmacist should be able to request for needed pharms, if they are absent or needed for making new medicaments Information that should be stored: - Medicines list - Medicaments prescription
Priority	2
Risk	L

Requirement ID	10.4
Title	System should allow Pharmacist to view patient's prescription
Type	Functional
Description	Pharmacist should be able to view patient’s prescription, understand what medicaments patient needs. Information that should be stored: - Patients account - Medicaments list
Priority	2
Risk	L

11. Warehouse

Requirement ID	11.1
Title	System should allow Warehouse to update a list of available inventory
Type	Functional
Description	Warehouse should be able to make changes in database, increase and decrease the amount of available inventory units, create new fields for counting new inventory, delete fields of old positions. Information that should be stored: - Inventory list
Priority	2
Risk	M

Requirement ID	11.2
Title	System should allow Warehouse manager to request financial support for inventory
Type	Functional
Description	Warehouse manager should be able to request for financial support for new inventory or for repairment of existing. Information that should be stored: - Inventory list
Priority	2
Risk	L

Requirement ID	11.3
Title	System should allow Warehouse manager to accept or deny requests
Type	Functional
Description	Warehouse manager should be able to accept or deny requests to get a new inventory. Information that should be stored: - Inventory list - Inventory cost - Description
Priority	3
Risk	L

12. Additional

Requirement ID	12.1
Title	System should be able to support “System integrity”
Type	Functional
Description	The system should ensure that only authorized users have access to the data. Information that should be stored: - Check login/password - Accounts information
Priority	1
Risk	M

Requirement ID	12.2
Title	The system should provide opportunity to manage user account invoice
Type	Functional
Description	The system should provide opportunity to manage user account invoice: - Create invoice - Pay invoice - Appeal invoice
Priority	1
Risk	M

Requirement ID	12.3
Title	System should be able to support “Additional staff”
Type	Functional
Description	The system should be able to include additional information about cleaning and technical service. Information that should be stored about a staff: - Name - Surname - Job title - Phone number - Salary
Priority	2
Risk	M

Non-functional requirements

Requirement ID	NF1
Title	System should be available during the weekday
Type	Non-functional
Description	The system should be at least 99.5% available between 9:00 a.m. and 6:00 p.m. on weekdays
Priority	1
Risk	H

Requirement ID	NF2
Title	System should support PGP encryption
Type	Non-functional
Description	Customer credit card information should be PGP encrypted by the system
Priority	1
Risk	H

Requirement ID	NF3
Title	System should be integrable with different software
Type	Non-functional
Description	The system should be compatible with browsers: - Based on Chromium - Safari
Priority	1
Risk	H

Requirement ID	NF4
Title	The system should provide suitable performance
Type	Non-functional
Description	Each request to the system should be processed in less than 5 seconds
Priority	2
Risk	H

Requirement ID	NF5
Title	System should be robustness to changes made by the user
Type	Non-functional
Description	The system should save every change made by user every 10 seconds.
Priority	3
Risk	L

Part 2

The ERD to Hospital Management system contains two types of accounts: **Employee** account and **Patients** one.

Due to the fact that each job has its own attributes and relations they were separated by entities: **Doctor**, **Nurse**, **Head of department**, **Pharmacist**, **Warehouse manager**, **Financial**, **IT specialist**, **HR**, **Security**, **Cleaning**. Each of such entities is connected with account by **has** relationship and are defined uniquely by their SSN.

Doctor. This entity has some special attributes like date of last training and status: are they on training now or not. Sending on training is an operation that can be performed by the **Head of Department**, where doctor works.

Department is a special type of entity due to the fact that it has separate management and rooms at the hospital. The head of department on ERD is not unique due to the fact that the hospital there also exists a head of the hospital, denoted as a head of all departments for simplicity of implementation.

Doctors have also got a relations with Patients - Appointments and Recipes. Appointments should be conducted somewhere that is why it is a ternary relation on Doctor, Patient and **Patients Timetable**, which is a collection of all available doctors with rooms, it can be seen by a patient. **Recipes** are also an important part of medical treatment. The job of theirs creation was delegated to the doctor due to the fact that some treatment could be harmful for people without some illness or can be used in non-appropriate way. However, there is also a **Medical Report** relation, which was delegated to the Nurse as a kind of routine work. Some drafts could be created by the doctor, but the last one shouldn't care about formality and the system work: doctor should think about the patient.

Medical staff should be able to communicate, don’t they? That is the reason, why in our diagram we have a **Message** relation from one employee to another. In addition to the message our system has more formal way of communication - **Requests**. There are a lot of them: request for vacation, request for pharms and inventory, even request for being fired. All these issues were generalized to the request relation, which will store the key information in a more efficient way, then the usual message does.

Request is only the request, but how does the **addition or firing** of employee happens? - Documents are prepared by HR and the Head of department made such

decision. That is the reason, why Addition and Firing is a ternary relation on HR, Head of Department and Account of Employee.

What about the financial part of the hospital? It is quite more complicated in terms of entities. The hospital has some invoices to pay to the systems outside of the hospital, while some money could be earned from the patients (e.g. for some procedures (appointments) or for medicines). Due to all these factors the **bill** is a special entity, not a relation between patient and financial manager. **Bill** has such attributes as from_id ad to_id, which let us to perform both invoice payment and money earning.

Pharmacist. This kind of specialist is quite separated from the other system, however he does a lot - he manages **pharms** (special entity because it can exist without a manager). If patient would like to buy them, he or she can directly request for them and the bill will be created. Such **payment** is a ternary relation, which connects patient, pharmacist and the new bill.

Also in the hospital management system there is a **warehouse manager,** who manages the **inventory** (separated entity) of the hospital and checks its state.

In addition, our system provides such services as **TODO-list**, **Staffs Timetable** and **Notice board** to the employees. Timetable is updated every month, while Notice board is preserved for a season. TODO-list is a weak entity, because it is defined by the person.

Let’s return to the patient. What if he or she has problems with a system? Such a situation could also happen with a doctor or nurse, that is why an IT-Specialist, who manages accounts exist and is open for communication by **contacting** relation.

The last thing in the system, which was not described yet is a **feedback**, that could be optionally left by some patient and is identified by the patient and the time when it was sent.

P. S. File with ERD is additionally attached to the submission

ERD of Hospital management system (Chen's notation)