



Epoka University
Faculty of Economics and Administrative Sciences
Business Informatics
CEN302 -- Software Engineering

E-DENT

Requirements Specification

Prepared by:

Blirona Keraj

Fiorela Marra

Inejda Pali

Tea Facja

Tomi Jorgji

Xhoana Hajdari

Received by:

Dr. Igli Hakrama

Table of Contents

1. EXECUTIVE SUMMARY.....	5
1.1 Project Overview.....	5
1.2 Purpose and Scope of this Specification.....	5
2. PRODUCT/SERVICE DESCRIPTION	5
2.1 Product Context	5
2.2 User Characteristics.....	6
2.3 Assumptions	7
2.4 Constraints	7
2.5 Dependencies	7
3. REQUIREMENTS	8
3.1 Functional Requirements.....	8
3.2 Non-Functional Requirements	13
3.2.1 Product Requirement.....	15
3.2.1.1 User Interface Requirements	15
3.2.1.2 Usability.....	16
3.2.1.3 Performance	17
3.2.1.4 System Interface.....	18
3.2.1.5 Security.....	18
3.2.1.6 Data Management	18
3.2.2 Organizational Requirements.....	20
3.2.2.3 Operational Requirements.....	20
3.2.2.4 Development Requirements.....	20
3.2.2.5 Environmental Requirements	21
3.2.2.6 Portability.....	21
3.2.3 External Requirements	21
3.2.2.1 Regulatory Requirements.....	21
3.2.2.2 Ethical Requirements.....	21
3.2.2.3 Accounting Requirements.....	22
3.3 DOMAIN REQUIREMENTS.....	22
4 SOFTWARE DESIGN / DIAGRAMS	23
4.1 User Scenarios.....	23
4.2 Use Cases.....	35
4.3 Diagrams	55

[E-DENT] Requirements Specification

1. Executive Summary

1.1 Project Overview

Describe this project or product and its intended audience, or provide a link or reference to the project charter.

Automatization and the decrement of the human intervention is basically the main course all businesses are following, to succeed in an environment where the customer satisfaction is crucial. This model, businesses are following, provides fast service, transparency, easy access by the customers and as a result more successful business and more loyal clients who are satisfied with the service.

A system for a dental clinic is what we are going to establish. Having a system to organize all the services and customers, helps the clinic to easier manage the operations, schedules, keep track of the patients cards etc. In other words be more efficient. Everyone who wants to receive a service from this dental clinic can book his appointment and the type of service needed, based on the available schedules. Another user of the system is the service providers, dentists. They can have their profile where will be stored all the patients' cards under their care. And the third user is the admin who will add or delete dentists', patients', and deal with inventory. Also there will be a user for receptionist who will deal with accounts keep track of the schedules and notify the patients in case of any change to the schedule or other operations.

1.2 Purpose and Scope of this Specification

Our purpose as the developers of this software is to improve customer experience and facilitate operations inside and outside of the dental clinic through E-DENT (Web Application). This software will provide very practical and simple solutions to the issues that the owner, patients, doctors and receptionists have addressed. Not only will E-DENT make it easier for potential patients to approach the clinic, but also for the clinic to be closer to the patients in case of any complication. Shortening the distance by creating a well-balanced frequent interaction between each of our users, we aim to facilitate each process from the attraction of the patient to the conversion in a promoter. The software is developed to be used by one dental clinic since we will customize it according to our client's requests and budget.

2. Product/Service Description

2.1 Product Context

E-Dent is a software which is designed to manage and facilitate the activities of a dental clinic. This product is a flexible and effective solution of managing jobs by converting medical and inventory data on paper into electronic files.

The software will be used by every employee of the clinic, but also by loyal and potential patients. Main goal of this platform is to enhance communication between staff and clients and help the staff coordinate their job and schedules in order to provide a satisfying user experience. Our software will be accessed by four users: patients, receptionist, dentists and admin/manager. The software aims to facilitate every step of clinic job from booking from

clients, to booking approvals, tracking dentists' timetables, generating patients' records, ordering from dental depots.

[E-DENT] Requirements Specification

The software will also serve as a website for potential customers who can navigate and see services and prices offered, experiences and results of other patients' from esthetic and surgical intervention, and contact for any information.

2.2 User Characteristics

The users of the system are :

Patients: anyone who wants to receive a service by this clinic can:

- register to the system to book appointments specifying the service based on the list (also prices) provided in the system ,
- have personal dental record section with previous appointments
- view different experiences that other clients had with this dental clinic and the dentists to whom they will trust their oral health
- book and cancel appointment

Dentists: Every dentist in the clinic will:

- be registered in the system
- keep track of the patients he has under care
- add and update dental records
- communicate and advise them when they have a concern
- view his schedules and appointments assigned to him
- declare the materials used which will be subtracted from the main inventory.

Receptionist: as a person who deals with the schedules and appointments he will:

- have a general calendar with appointments organized in the system and directly updated by the patients actions
- add/edit the appointment scheduled
- cancel appointment
- assign dentists for each appointment taking into consideration their workload
- view each dentists' schedules
- communicate it with the patient using the contact information in the patient records, in case of any changes in schedules
- keep track of the payment completion of the services provided, indicated by the Paid/Not Paid status for each patient.
- Register the patients in the system if they directly request service in the clinic

Admin: as the main actor of the system will:

- diss/approve the registration of the dentists
- be able to change the list of the services according to the decisions made for expanding/shrinking the services or changing the prices
- Be able to generate financial reports

Economist: the person responsible for the internal and external financial transactions

- keep track of salaries payments
- keep track of payments for each service provided by the clinic
- keep track the inventories of the clinic and make sure the needed materials are ordered
- keep track of the payments made for each new order
- keep track of any other expense
- preparing monthly reports of the cash flow

2.3 Assumptions

The list of assumptions :

- It is assumed that each user has limited working proficiency in English.
- It is assumed that each patient in cases when is registered by the receptionist gives the correct personal information and the approval for the account registration.
- It is assumed that the admin account is created by the system developers and given to the clinic's owner/ IT manager.
- It is assumed that the clinic and the patient own smart devices and have internet connection to be able to access the system.
- It is assumed that the system is built in accordance with the Albanian Law No.9887 "ON PROTECTION OF PERSONAL DATA"
- It is assumed that every patient should be limited to watching only his booking and not interfere with other patients.

2.4 Constraints

List of constraints:

- The internet connection when absent or not powerful enough can be a constraint for the fully functioning system.
- It is required that every device should be equipped with browser to run the application
- Budget of the client.
- MySQL will be used for fetching data from database which may not be very efficient for large database.
- For very large databases the management efficiency by the SQL may fade.
- The system will only use the data contained in the existing corporate database.
- The system shall be available 99.99% of the time for any 24-hour period.

2.5 Dependencies

The list of dependencies that affect the requirements:

- The willingness of patients to use the system and not to request information manually from the receptionist
- the patients and clinic's staff access to the internet
- Depends on successful communication with related dental depots. This is achieved by maintaining an accurate inventory, by generating weekly reports on the quantity of used and available products.
- The execution of the sql queries performed by the users through the software's interface are dependent on the capacity of the database , the way the entities are organized and the number of users concurrently using the system.

3. Requirements

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_01	The system is implemented as a web application used by four functional user modules: admin, patients, dentists, receptionist.	This will be the main platform of operation for all users	1	27.05.2020	Blirona Keraj
BR_02	The administrator account must have the majority of the privileges of the system, and be able to manage the main operations provided by the data available on the system.	The system is under the management of the admin who can edit and manage the users and operation related to the system.	2	27.05.2020	Blirona Keraj
BR_03	The system should associate a supervisor indicator with each user.	The admin can have under supervision al the users accounts and have full access on them	3	27.05.2020	Blirona Keraj
BR_04	The system must provide the admin account with the competence to control the creation of the users account except the patients' one.	The admin can use a button to approve or disapprove the creation of a new employee account(dentist,receptionist).	2	27.05.2020	Blirona Keraj
BR_07	The admin can log in into his account using the credentials received by the development team.	The admin receives a username and password by the system developers and use to log in.	1	27.05.2020	Blirona Keraj
BR_08	The system must give the opportunity to the admin to change the credentials.	Using the change credentials button admin can change the username and password.	2	27.05.2020	Blirona Keraj
BR_09	The system must provide graphical data about the inventory to allow Economist be able to manage and meet the demand	This will be important to have a clear idea about the stock material to help the decision making related to inventory.	3	27.05.2020	Blirona Keraj
BR_10	The system must provide the option to add the dental materials specifying the name, the quantity and the price.	The Economist will frequently update the inventory database to have the accurate quantity of materials.	2	27.05.2020	Blirona Keraj

[E-DENT] Requirements Specification

BR_11	The system must provide the option to remove from inventory the used up material according to the name and specifying the quantity.	The Economist can use remove command to exclude the used materials from the inventory to have updated quantity.	1	27.05.2020	Blirona Keraj
BR_12	When a specified min quantity is reached the system must notify the Economist for refurbishment.	The clinic will always have the necessary material on time as the Economist can order them as soon as notification appears.	3	27.05.2020	Blirona Keraj
BR_13	The system must allow the admin to edit the list of the services offered by the clinic.	The admin can add/delete service types meeting the patients demand and clinics decisions about the progress.	2	27.05.2020	Blirona Keraj
BR_14	The software must allow the admin to Edit the prices specifically for every service type in the list.	The admin can change the prices according to any discount or other decisions made by the owner.	2	27.05.2020	Blirona Keraj
BR_15	The software must allow the admin to delete an employee account.	In case a dentist or receptionist leave or is fired the admin must remove his account from the system	2	27.05.2020	Blirona Keraj
BR_16	Each user should be identified by unique username and valid password.	No two users have the same username and password is hashed before stored in the database.	1	27.05.2020	Blirona Keraj
BR_17	Admin should have restricted access to users' sensitive information.	Admin should be able to change only his password, not the others.	1	27.05.2020	Blirona Keraj
BR_18	Dentists should be able to cancel their appointments	If for any personal reason, the dentist can not reach the appointment, he has the right to cancel it.	2	27.05.2020	Blirona Keraj
BR_19	Patients should be notified by email in case that appointment is canceled.	An email is automatically sent to the patient to notify that his/her appointment is canceled.	3	27.05.2020	Blirona Keraj
BR_20	Patients can cancel their appointment and book another one.	If the patient can not attend the appointment to the clinic for any reason, can cancel it.	3	27.05.2020	Blirona Keraj

[E-DENT] Requirements Specification

BR_21	Receptionist and dentists are notified for the cancellation of an appointment by a patient.	A notification is automatically sent to the receptionist and dentists if there is any change in bookings.	3	27.05.2020	Xhoana Hajdari
BR_22	Software should be able to generate XML/PDF files which store all records of a specific patient.	Each patient has its own medical records which are stored and updated in the system.	3	27.05.2020	Xhoana Hajdari
BR_23	Everyone can look at the clinic webpage and get information	Everyone who has a browser can access the webpage and find information regarding services, dentists, address.	1	27.05.2020	Xhoana Hajdari
BR_24	The system shall provide the user “Patient” with the option of booking appointments.	This option will make it possible for the user to check availability and make a decision on the booking date and time	1	27.05.2020	Xhoana Hajdari
BR_25	The main system page, the home page shall give general information about dentists, reviews, contact details, location, services.	.In order to have a clear idea about the clinic and its type of services offered a home page providing all of these is created	2	27.05.2020	Xhoana Hajdari
BR_26	The system shall allow the user “Patient” to chat with his or her own doctor.	This option will make it possible for the user to consult with the doctor in case of any complication or situation that might occur.	3	27.05.2020	Xhoana Hajdari
BR_27	The system shall provide the user “Patient” with full access to own dental records in the clinic.	Will make it possible for the user to check each service that has been provided to him and the advancements that he/she has had since the treatment.	2	27.05.2020	Xhoana Hajdari
BR_28	The system shall include detailed information about the user “Patient” and a dental record on all the services received at the clinic.	It will be stored in the database and the users will have limited access on it except the admin.	1	27.05.2020	Xhoana Hajdari
BR_29	The system shall provide the user “Patient” with the list of services offered in the clinic and the respective prices.	Will make it possible for the user to know what actually is offered in the clinic and be aware of the amount the clinic will charge them.	2	27.05.2020	Xhoana Hajdari
BR_30	The system shall allow the user “Patient” to edit all the attributes in his/her account, except the ones defined by the admin.	Each user shall be able to edit the specified features in the personal account.	2	27.05.2020	Xhoana Hajdari

[E-DENT] Requirements Specification

BR_31	The system shall provide the user “Patient” with his/her personal account.	Each user shall have an account with his/her own personal data.	1	27.05.2020	Xhoana Hajdari
BR_32	The system shall allow the user “Dentist” to chat with his or her own patients.	This option will make it possible for the user to check up with the patients in case of any complication or situation that might occur.	3	27.05.2020	Xhoana Hajdari
BR_33	The system shall provide the user “Dentist” with dental records on all patients that received services at the clinic.	Will make it possible for the user to check each service that has been provided to the patients in the clinic.	1	27.05.2020	Xhoana Hajdari
BR_34	The system shall provide the user “Dentist” with the complete list of scheduled appointments assigned him/her.	Will make it possible for the user to check on his availability and also know what is on his agenda.	2	27.05.2020	Xhoana Hajdari
BR_35	The system shall provide the user “Dentist” with his/her personal account.	Each user shall have an account with his/her own personal data.	1	27.05.2020	Xhoana Hajdari
BR_36	The system shall allow the user “Dentist” to edit all the attributes in his/her account, except the ones defined by the admin.	Each user shall be able to edit the specified features in the personal account.	2	27.05.2020	Xhoana Hajdari
BR_37	The system shall provide the user “Dentist” with limited access to the information about the users.	The user can not have access to every data on the database.	3	27.05.2020	Xhoana Hajdari
BR_38	The system shall provide the user “Dentist” with full access to his/her own profile.	The user can have access only to the specified data data on the database.	2	27.05.2020	Xhoana Hajdari
BR_39	The system shall provide the user “Dentist” with full access to patients’ dental records.	The user can have access only to the specified data data on the database.	2	27.05.2020	Xhoana Hajdari
BR_40	The system shall provide the user “Receptionist” with his/her personal account.	Each user shall have an account with his/her own personal data.	1	27.05.2020	Xhoana Hajdari
BR_41	The system shall allow the user “Receptionist” to edit all the attributes in his/her account, except the ones defined by the admin.	Each user shall be able to edit the specified features in the personal account.	2	27.05.2020	Fiorela Marra

[E-DENT] Requirements Specification

BR_42	The system shall provide the user “Receptionist” with limited access to the information about the users.	The user can not have access to every data on the database.	2	27.05.2020	Fiorela Marra
BR_43	The system shall provide the user “Receptionist” with full access to his/her own profile.	The user can have access only to the specified data data on the database.	1	27.05.2020	Fiorela Marra
BR_44	The system shall provide the user “Receptionist” with full access to patients’ dental records.	The user can have access only to the specified data on the database.	2	27.05.2020	Fiorela Marra

BR_45	The system shall provide the user “Receptionist” with full access to patients’ dental records.	The user can have access only to the specified data data on the database.	2	27.05.2020	Fiorela Marra
BR_46	The system shall provide the user “Receptionist” with full access to patients’ personal account.	The user can have access only to the specified data on the database.	2	27.05.2020	Fiorela Marra
BR_47	The system shall allow the user “Receptionist” to edit patients’ personal data.	Each user shall be able to edit any of the user “Patient” attributes in the respective profiles.	3	27.05.2020	Fiorela Marra
BR_48	The system shall allow the user “Receptionist” to delete patients’ personal data or account.	Each user shall be able to delete any of the user “Patient” accounts.	2	27.05.2020	Fiorela Marra
BR_49	The system shall allow the user “Receptionist” to add a new patients’ personal account.	Each user shall be able to add a new user “Patient” account.	2	27.05.2020	Fiorela Marra
BR_50	The system shall allow the user “Receptionist” to add a new appointment.	Each user shall be able to add a new appointment for any patient.	1	27.05.2020	Fiorela Marra
BR_51	The system shall allow the user “Receptionist” to edit an appointment.	Each user shall be able to edit an appointment for any patient.	2	27.05.2020	Fiorela Marra
BR_52	The system shall allow the user “Receptionist” to delete an appointment.	Each user shall be able to delete an appointment for any patient.	1	27.05.2020	Fiorela Marra
BR_53	The system shall provide user “Receptionist” with access to the user “Dentist” schedule.	Each user shall be able to see the agenda of each user “Dentist” in order to assign appointments properly.	2	27.05.2020	Fiorela Marra
BR_54	The system shall allow user “Receptionist” to edit the user “Dentist” schedule.	Each user shall be able to modify the agenda of each user “Dentist” in order to assign appointments properly.	2	27.05.2020	Fiorela Marra

[E-DENT] Requirements Specification

BR_55	The system shall provide user “Receptionist” with access to the user “Patient” schedule.	Each user shall be able to see the appointment of each user “Patient” in order to assign dentists correctly.	2	27.05.2020	Fiorela Marra
BR_56	The system shall allow user “Receptionist” to edit the user “Patient” schedule.	Each user shall be able to modify the appointment specifics of each user “Patient” in case of any change.	2	27.05.2020	Fiorela Marra
BR_57	The system shall allow user “Receptionist” to assign appointments to the user “Dentist” schedule.	Each user shall be able to assign an appointment by a specific patient to the available dentist.	1	27.05.2020	Fiorela Marra
BR_58	The system shall allow user “Receptionist” to assign dentists to the user “Patient” appointments.	Each user shall be able to modify the appointment specifics of each user “Patient” in case of any change.	2	27.05.2020	Fiorela Marra

BR_59	The system shall provide user “Receptionist” with the list of all appointments and their specifics.	All the users shall see all the appointments so there are no inconveniences nor confusions between patients and the dentists.	1	27.05.2020	Fiorela Marra
BR_60	The system shall provide the user “Economist” with his/her personal account.	Each user shall have an account with his/her own personal data.	1	27.05.2020	Fiorela Marra
BR_61	The system shall allow the user “Economist” to edit all the attributes in his/her account, except the ones defined by the admin.	Each user shall be able to edit the specified features in the personal account.	2	27.05.2020	Fiorela Marra
BR_62	The system shall provide the user “Economist” with limited access to the information about the users.	The user can not have access to every data on the database.	2	27.05.2020	Fiorela Marra
BR_63	The system shall provide the user “Economist” with full access to his/her own profile.	The user can have access only to the specified data on the database.	2	27.05.2020	Fiorela Marra
BR_64	The system shall provide the user “Economist” with full access to patients’ payments list.	The user can have access only to the specified data on the database.	1	27.05.2020	Fiorela Marra
BR_65	The system shall provide the user “Economist” with limited access to dentists’ profile.	The user can have access only to the specified data data on the database.	2	27.05.2020	Fiorela Marra
BR_66	The system shall allow the user “Economist” to edit dentists’ payment records.	The user shall be able to edit that attribute of the user “Dentist” in the respective profiles.	1	27.05.2020	Fiorela Marra
BR_67	The system shall allow the user “Economist” to edit patients’ payment records.	The user shall be able to edit that attribute of the user “Patient” in the respective profiles.	1	27.05.2020	Fiorela Marra

[E-DENT] Requirements Specification

BR_68	The system shall allow the user “Economist” to edit receptionist’ payment records.	The user shall be able to edit that attribute of the user “Receptionist” in the respective profiles.	1	27.05.2020	Fiorela Marra
BR_69	The system shall allow the user “Economist” to edit admin’ payment records.	The user shall be able to edit that attribute of the user “Admin” in the profile.	1	27.05.2020	Fiorela Marra
BR_70	The system shall provide the user “Economist” with full access to the inventories database.	The user can have access only to the specified data on the database.	1	27.05.2020	Fiorela Marra
BR_71	The system shall allow the user “Economist” to edit the inventories payment records.	The user shall be able to edit that attribute in the database.	1	27.05.2020	Fiorela Marra

3.2 Non-Functional Requirements

3.2.1 Product requirements

3.2.1.1 User Interface Requirements

- E_DENT is going to be a web-based application, which can be accessed at any browser including Google, Mozilla, Safari, Internet Explorer.
- The user interface of the system will be user-friendly, easy to use, supported by the overall organized structure of the modules and not a complex frustrating one. No manual or training will be necessary to be able to use this system, however a descriptive document with the system functionalities will be provided.
- The main interface all the user will be presented to will be the home page divided into different sections redirecting to specific pages. Into the home page there will be the option for the login, register, Dental services, Smile Gallery, Meet our Dentists, About us. A side bar for easier navigation will be on the right side of the interface.
- The register module will have a form where the required information will be filled to later be sent to the database. Some of the required data may have some constraints such as password format. For any non compatibility with the format, specified also close to every input field, an alert will be view on the screen warning the user for the errors to make it easier and more transparent experience.
- The log in module uses the credentials input for each user to connect to the database, authenticate and provide access to each specific account. When a non-existent user or a wrong password is typed a small alert will be shown to notify the user for this error.
- The admin user will initially use the credentials received by the developing team to log in into his account. There will be the Inventory, Edit Services and the HR section. Each of them located on top of the page will be accessed through the on click command.
- The Inventory section will provide graphical data generated by the data of the current items of the inventory. This view will give the admin the general insights and an easier way to manage inventory. The table with the items will be shown and the button for removing or adding new item materials.
- The HR module available only for the admin will include the list of all employees and the functionalities to add/remove employees and the wages records. A notification will be provided by the interface for any latency in wage delivery.
- The Edit Services module will be plain and simple including the current list of the services and prices and the edit button adjacent to each of them.
- The interface of the module accessed by the receptionist will contain a calendar of all the appointments booked by the patients specifying the ones with assigned dentists or not. Also the receptionist will be able to view the Agenda of the specific dentists and interact with the page by assigning dentists to the appointments according to free hours or days shown.
- The receptionist module will also provide a table with the payments of the services offered with the feature of the Paid/Not Paid status for each service provided. In the cases when the patients records are needed at the moment the receptionist can view them from the Patients' Records section.
- The dentist after he logs in he will be directed to his account where a calendar of his agenda will be shown. The interface will show the notification with a red dot on top right corner for the new

[E-DENT] Requirements Specification

appointments assigned by the receptionist. Also the dentist will be able to view and edit the Patients Records(cards).

- The patient after they get a brief information on the home page and decide to open an account after they log in they will see the list of the upcoming scheduled appointments(if any). Also one of the section on patients module is the Patients personal Record. The interface will also provide with a notification signaling new messages from the Consult with Your Dentist section.
- The patient can view all the services and prices and also on the Book an Appointment part a calendar with free spots will be shown to make it easier for the patient to plan the next appointment.
- The overall user interface will provide an understandable design and structure with a lot of functionalities that make the system highly interactive and organized.

3.2.1.2 Usability

Software is user-friendly which means that is very easy to work with and each user will find it easy and comfortable to access it. Users will learn how to use the software application within a couple of minutes as every step is easily identifiable through design in order to provide a satisfying user experience.

Instruction and alert messages are shown during the execution of the task. For example if a user tries to log in with a wrong username or password, it shall display “Wrong username or password, please try again”. Icons and menus facilitate the user to perform specific tasks.

Accessibility

- Since it is a Web application it will be accessible on any device, at any time only if the user has access to the internet. Software is accessed by four users: admin, dentists, receptionist, patient. These users can access the software through their accounts. The Website is also available to everyone who is interested in the clinic services and can be accessed through a web browser such as Google Chrome or Mozilla.

Responsiveness

- The software should be highly responsive in design, data transactions and social network that will be implemented as part of the web to provide satisfying user interactions.

Flexibility

- The software should be constructed in such way that it is easy to be modified and updated in order to accommodate new requirements.
- The software application should be designed in such manner that is able to isolate and manage errors efficiently. In case that user enters wrong data input, the system shall generate alerts or ask for confirmation to proceed.

Effectiveness

- The software application will be easy to use and learn by all users. A manual in English and Albanian language will be provided to facilitate the users to efficiently access the system.
- The software will provide both staff members and patients with practical tools of managing their data and with a convenient way of performing their tasks across the platform.

Efficiency

- The software system is designed to be fast, efficient and reliable.
- The software will provide users and administrators with a fast and reliable way of accomplishing their jobs such as creating and scheduling appointments or updating dental information in little time at their own convenience.
- The design will ease the use of interface as the icons, buttons, headings will make it easy for any user to perform any operation.
- In case that a user enters data input incorrectly, the system will check and warn the user.

3.2.1.3 Performance

- It will be a web-based software, so it is required to be run from a web server.
- The performance of this app is strongly determined by the user's internet connection strength, server hardware performance, the algorithm's efficiency on fetching the necessary data from the database, the number of active users that are accessing the web at the same time and on the operating system that is installed in the server.
- Powerful server machines with high band internet access are needed to provide a better performance and handle multiple users.
- A high storage space is needed to support more users and bigger workspace per user.
- The web application should be developed as a lightweight web app so that it can work on almost any platform even with slower internet connections.
- The software shall support all the users who must have access in the system at any time.
- The software will take initial load time depending on internet connection strength which also depends on the media from which the product is run.
- The performance will depend upon hardware components of each user. Registration of data for each entity shall be processed in a few milliseconds.
- The software shall support 2000 users a day at maximum traffic.
- Software Response Times for failure should meet 95% of all pages in less than 8 seconds.
- DB system should easily handle 10,000 read queries per hour but only 3,000 update operations per hour.
- The system should be able to process 50 queries per second.

Capacity

- Average page load (from a user perspective) must be less than 500 milliseconds.
- Slowest page load cannot take more than 4 seconds.

Availability

- The app must be available more than 99.5% of the time, on average.
- The app must be available more than 99.999% of the time during the hours of 8am-8pm GMT+2.
- The app must be available and accessible by any user worldwide.

Latency

Latency is a time interval between the stimulation and response. Average speed for the completion of a full interaction is estimated to be faster than 50 ms.

3.2.1.4 System Interface/Integration

The database of our system will be provided to the dentists, receptionist and economist, as information. Not all of them will be able to make changes. Only the administrator can make changes to the structure of the database.

Network and Hardware Interfaces

The system will be a web application stored in web server, this means that a TCP connection will occur between the server and the browser. Every browser will be able to support the connection, this way our system will function properly and the administrator, the receptionist, the dentists and every patient will be able to access the system when in the right conditions provided.

3.2.1.5 Security

- In this system will be stored these types of data: Personal data of each user, Clinical Records of patients, Credentials of each person. Under the privacy policy, the dataset will be encrypted, safe from unauthorized usage and stored for unlimited periods of time.
- The methods that will be used to insert and store data in the database, will assure stability, check data integrity and prevent injections from inside or outside of the system.
- We will use methods as sanitizing, hashing and encryption in order to protect our system from external hazards.

Authorization and Authentication

- New users will be registered with a username and password. At that time an email with a randomly generated password that is automatically encrypted with b-encryption standard shall be sent to the .
- The database is centralized to a single admin user and accessible only to the administrator account who performs all the CRUD functionalities on the data

3.2.1.6 Data Management

The database will hold various types of data necessary for a fully functional system. The connection between php functions with the database make it possible for the users to interact with the data through the interface. The access to this data will depend on the privileges and scope of functions of each user. The data is organized in entities which are connected with each other by relationships. These relationships create the data flow model among the users. These users will implement these relationships due to different features and functions that provide the interactivity with the data of the system. The overall structure of the entity relationships of the database will be build to be efficient in responding to users commands

- Almost all of the data will be generated by the users as they continue to use the system, except for some specific data that has to be initialized like the credentials of the admin(editable).
- The system's database will have data regarding the employees including their personal information, the patients they have under care and the wages amount and delivery time. This information is accessed and manipulated only by admin and the employees but only for their account,

[E-DENT] Requirements Specification
not for the other employees

- The database will hold the data regarding the inventory items and quantity, name and price for each. This data is accessed and manipulated only by admin.
- Because the data validations are performed during input in forms of registration and log in and so on the accuracy of the data is relatively high. A special attention must be paid to the quantities and prices as the information they contain is important for the clinic.
- The system will contain data for the service list and the prices for each. This data can be viewed by everyone but can be edited only by admin. This type of data may be most frequently used by the patients to book appointments.
- The database will hold data type photo(file) which are part of the Smile Gallery section. This data is uploaded by the admin into the database and can be accessed and viewed by everyone.
- The system's database will also contain all the patients' records which are input of dentists. This data can be one of the most frequently used data as the patients supervision is one of the core functions of the clinic.
- One of the main and most frequently used data is the one describing the appointments of type Date which is connected with dentists and service types.

3.2.2 Organizational Requirements

3.2.2.1 Operational Requirements

The E-DENT online software has the main purpose of facilitating the interaction between the dental clinic and its clients, while:

Allowing clients to have a personalized profile where they are provided a set of operations:

- Have editing permission of their profile personal information (Create, Read, Update, Delete)
- Set an appointment to the clinic
- Have an accessible personalized history of their dental records
- Give feedback of the service via rating and/or comments
- Have access on previous feedbacks of different clients

Giving the receptionist the responsibilities of

- Maintaining the timetable by CRUD functionalities over appointments(appointing dentists and time) as well as approving them
- Having CRUD functionalities over clients profiles when having clients verbal consent
- Having the responsibility of checking and editing clients payment status

Dentists operations consist of:

- CRUD access of their profile
- Read-only access to his/her assigned clients dental record
- Declaration of used material on each appointment by ticking boxes
- Access on his/her personal timetable

Admin operations consist on:

- CRUD functionalities of dentist and receptionist profiles
- CRUD functionalities on the menu of services and prices
- CRUD functionalities on the list of inventory materials
- Access the workers payroll, and expenses
- Generate reports regarding expenses, income, payroll.

3.2.2.2 Development requirements

Front-end - Client-side programming

- HTML (Hypertext Markup Language) and CSS (Cascading StyleSheet), used through Microsoft Expression Web 4 framework
- JavaScript with JQuery libraries to make the page interactive

Back-end - Server-side programming

- PHP programming language through the best free framework (Laravel)
- MySQL database for data storage
- VPS (Virtual Private Server) Hosting for handling clients requests

3.2.2.3 Environmental requirements

- Physical space of workstations places in every examination room as well as in the receptionist desk and main office
- Power supply through normal infrastructure which is connected to power generator in case of energy blackout.

3.2.2.4 Portability

- Since the software is a web based application, it can operate in any device and be accessed by anyone who has internet connection and a web browser.
- Only staff members and patients who have an account and password to log in can collaborate with other members and execute operations in the system.
- Real time updates when bookings and inventory orderings are made.
- The used portable scripting language will be php.

3.2.3 External Requirements

3.2.3.1 Regulatory Requirements

The system will operate under a specific address that we will create and provide further in the project. Privacy policies will be in compliance with the Law No. 9887, dated 10.03.2008, “ON PROTECTION OF PERSONAL DATA” and related sub-legal acts. Privacy Policy describes and includes the types of collected information and also the way of usage of this information. As most entities on the Internet, visitors’ IP addresses to the web-based application are used to help diagnose problems with the server. IP addresses are not linked to anything personally identifiable, so the user sessions will be tracked, but they will remain anonymous.

Also, according to the law No. 9887, dated 10.03.2008, as amended with law No.48/2012, “On Protection of Personal Data”, the personal information of each user should be private and accessible only by the specific people.

This web-based application will also enforce the latest update on the General Data Protection Regulation (GDPR) as it became enforceable on May 25, 2018 in EU (European Union) and EEA (European Economic Areas) countries.

3.2.3.2 Ethical Requirements

The personally-identifiable information voluntarily provided by the users, such as name, surname, date of birth, phone number, email address will not be traded to other parties or shared. Controllers, processors and persons who come to know the content of the processed data while exercising their duty, shall remain under obligation of confidentiality and credibility even after termination of their functions.

At the same time, personally-identifiable information will be provided to third-parties if one or more of the following conditions apply:

- The administrator has received the consent of users to share the information.
- The administrator is requested or authorized to share the information by the legislation in force, court order or an act of a public or regulatory authority.
- The administrator shares the information in order to protect specific rights or staff members.

3.2.3.3 Accounting Requirements

Accounting requirements will be in compliance with the Law No. 25/2018 on Accounting and Financial Statements that was approved on 10 May 2018 by the Albanian Parliament abrogating Law No. 9228 dated 29 April 2004 on Accounting and Financial Statements.

Public interest entities that exceed the criteria of medium enterprises and the regulators of public interest entities (i.e. the Bank of Albania and Financial Supervising Authority) will apply the International Financial Reporting Standards (IFRS).

All other enterprises will apply the National Accounting Standards (NAS). Entities may also select voluntary application of the IFRS.

3.3 Domain Requirements

Domain requirements reflect the environment in which the system operates, in our case a system for a dental clinic. The main purpose of this platform is to extend up the communication between clients and staff and allowing clients to have a personalized profile where they are provided a set of operations.

Our software will be only accessible by five users which are patients, receptionist, dentists, manager(admin) and economist. The application is developed to be used by one dental clinic in order to customize the client's budget and their requests.

4. Software Design / Diagrams

4.1 User Scenarios

Admin fails to log in:

1. The log in form to input the credentials is shown
2. Admin types the username and password incorrectly
3. The alert is shown for the incorrect credentials pop-up
4. The message for log in failure shows
5. The re-input of the credentials is requested

Admin successfully logs in:

1. The log in form to input the credentials is shown
2. Admin types the username and password correctly
3. After the database check is done the admin is directed to his homepage
4. The admin views his homepage

Admin changes credentials and profile info:

1. On admin's homepage enter on the profile section
2. The profile page is viewed with the main information and the username and password
3. Use the edit functionality to make all textfields editable
4. Make the desired changes according to the validation rules and save changes
5. A message confirming the successful change appears

Admin fails to change credentials:

1. On admin's homepage enter on the profile section
2. The profile page is viewed with the main information and the username and password
3. Use the edit functionality to make all textfields editable
4. Make the desired changes violating the validation rules and try to save them
5. Alerts on each text field show indicating the unsuccessful procedure

Admin edits the employee's list:

1. On the homepage access the Employee section
 2. A table on all employees divided by category is showed
- Adds new employees:
 1. Use the functionalities to Add an employee
 2. A form with required information is required to be filled
 3. Admin fills it and saves the operation
 - If info is filled according to the validation rules the message indicating a successful process shows
 - If info is filled violating the validation rules alerts on each text field for the error show and a message informing that the process failed
 - Admin removes employee from the list:
 1. On the action section of the employee's list use the feature of removing the employee he wants to remove
 2. A reconfirmation message appears "Do you want to remove this employee?"
 3. Approve it , the message indicating a successful process shows

[E-DENT] Requirements Specification

- Admin view employee's profile:
 1. On the action section of the employee's list use the functionality to view the full profile on the specific employee's row
 2. The profile with specific details is viewed
- If no changes are needed cancel the process
- return to the homepage

Admin Edits the Service List:

1. Enter on the Dental services section
2. A list of all Services and their prices shows
 - Admin changes the price/name
 1. Adjacent to each service use the feature for editing, the name and the price fields become editable
 2. Make the changes and save them
 3. The list is automatically updated
 - Admin removes the service from the list:
 1. Adjacent to each service there is a functionality that allows to remove Employees, use it
 2. A reconfirmation message appears asking approval to continue with the changes
 3. approve it and a message will inform you for the success of the procedure
 - Admin adds a new service in the list:
 1. Use the feature to add Services
 2. A row on the list with required details to be filled is added
 3. Fill and save , the new service is automatically part of the list
 - If no changes are needed Cancel the process
 - Return to the homepage

Admin checks the job applications:

1. On the homepage there is a section with most recent job application
2. Access the rest of the list
3. For each application view the full information and to give the status rejected or approved to the application

Admin requests Reports from Economist:

1. Access the Reports section on the homepage
2. Sections regarding the types of report to be requested shows
 - Admin requests financial reports:
 1. Access the "Financial Reports " and a number of option to chose is offered appears
 2. ""Balance Sheet", "Wages Schedule", "Income Statement", "Cash Flow" Is listed with a request action adjacent to each one of them
 3. perform request on the specific report and the request is sent automatically to Economist
 - Admin requests Inventory Records:
 1. Access the "Inventory Records "
 2. A request command shows
 3. perform request and the request is sent automatically to Economist
 - Admin communicates with Economist:
 1. If specific information is needed Admin can use the features to send comments or messages to the Economist
 2. The message is sent to the Economist

Admin views the received reports:

1. Enter in the "Reports" section on the homepage

[E-DENT] Requirements Specification

2. A Received Reports list is showed
 - Admin views financial reports:
 1. Access the “Financial Reports ” and a number of option to chose is offered appears “Balance Sheet”, “Wages Schedule”, “Income Statement”, “Cash Flow”
 2. Is listed with a open action adjacent to each one of them
 3. perform open function on the specific report and the report is provided to Admin
 - Admin views Inventory Records:
 10. Access the “Inventory Records ”
 11. A open command shows
 12. open the report file

Admin checks the messages received:

1. Use the interface to enter to the messages section
2. A list of all messages appears

Admin logs out:

1. Use the specific option to log out
2. The admin is sent back to the log in form page.

Successful Log in

- Receptionist enters the username on the specific required field.
- Receptionist enters his/her password.
- If the credentials are correct, match those saved in system database then the Receptionist is Logged in.
- The Receptionist is directed to the his Page, that provide all the Receptionist functionalities:

-His profile

- Appointments and Schedules

-Dentists’ Page

-Patients’ Page

Unsuccessful Log in

- Receptionist enters the username on the required field.
- Receptionist enters his/her password.
- If the credentials do not match those saved in system database, then an error message is displayed.
- Receptionist remains in the Log In page.
- Receptionist must re-enter his/her credentials.

Receptionist- Update Profile

[E-DENT] Requirements Specification

- Receptionist is logged in the system
- Receptionist goes to Profile section of his page.
- A form will be displayed with his registered information.
- Receptionist can change his information/picture.

Receptionist- Add a patient account

- Receptionist is logged in the system
- Receptionist goes to the Patient Page.
- Receptionist fills a form with needed information to be provided, regarding name, surname, username, phone number, age, password, category.
- If data input entered conforms with the system restrictions, then the user is successfully added.
- User is created as an entry in the Patient table of the clinic system database.

Receptionist- Delete a patient account

- Receptionist is logged in the system
- Receptionist goes to the Patient Page.
- Receptionist is shown a list of all patients of the clinic.
- Receptionist can delete a Patient from the list.
- User is deleted from the Patient's table entries of the clinic system database.

Receptionist- Searches a patient in the list

- Receptionist is logged in the system
- Receptionist goes to the Patient Page.
- Receptionist is shown a list of all patients of the clinic.
- Receptionist searches by typing the name of the patient he needs.
- List of patients is updated every time the Receptionist enters a key, to display the results.
- Receptionist is directed to that patient page.

Receptionist- View dentist schedule

- Receptionist is logged in the system
- Receptionist is directed to his/her page.

[E-DENT] Requirements Specification

- The Receptionist goes to Appointment and Schedules section part of his/her page.
- Receptionist can see the list of all appointments assigned.

Receptionist- Add Appointment & Schedules

- Receptionist is logged in the system.
- Receptionist is directed to his/page.
- Receptionist goes to Appointment & Schedules.
- The Receptionist is displayed a table with a list of appointments scheduled.
- Receptionist can add Appointment.
- Receptionist should complete a form with needed information to be provided: date, time, patient information, dentist, service to be provided.
- A new entry is created in the Appointments table of the clinic system database.

Receptionist- Cancel appointments

- Receptionist is logged in the system
- Receptionist is directed to his/page.
- Receptionist goes to Appointment & Schedules.
- The Receptionist is displayed a table with a list of appointments scheduled.
- Receptionist can edit and Cancel appointment of a specific patient from the list of appointments.
- A table pops in to confirm the cancellation of the appointment.
- If Confirm, the page is refreshed to show the updated list.
- Entry is deleted from the Appointments table of the clinic system database.
- An email is automatically sent to the respective dentist and patient to notify that his/her appointment is cancelled.

Receptionist- Checks a Patient Dental Record

- Receptionist is logged in the system
- Receptionist goes to the Patient Page.
- Receptionist is shown a list of all patients of the clinic.
- Receptionist can access a patient profile from the Patient list.

[E-DENT] Requirements Specification

- Receptionist can access Dental Records of the patient.
- The receptionist it is shown a list of dental record in PDF version.

Receptionist- Checks a patient payment

- Receptionist is logged in the system
- Receptionist goes to the Patient Page.
- Receptionist is shown a list of all patients of the clinic.
- Receptionists can access a patient profile from the Patient list.
- Receptionist can access the payment status of the patient.
- The receptionist it is shown the payment status: Paid/ Not Paid.

Receptionist- Checks a patient appointment request

- Receptionist is logged in the system
- Receptionist is directed to his/page.
- Receptionist can receive notifications for appointment request.
- Receptionist can view the patient's request in the notification window.
- Receptionist assigns appointment according to dentist's schedule and patient requirement.

Receptionist- View Appointment & Schedules

- Receptionist is logged in the system
- Receptionist is directed to his/page.
- Receptionist can access Appointment & Schedules section.
- The Receptionist is displayed a table with a list of appointments scheduled.
- Each entry in the table provides information such as date, time, patient name, surname, dentist, service, payment.

Successful Log in:

- Dentist enters the username on the specific required field.
- Dentist enters the password on the specific required field.
- If the required entered information is correct, matches that saved in the system database, then the Dentist is successfully logged in.

[E-DENT] Requirements Specification

- The Dentist is logged into his/her page, where the following options are provided on the right side of the page:
 - His/her profile - Patients' Dental Records - Appointments and Schedule - Messages - Log out

Unsuccessful Log in:

- Dentist enters the username on the specific required field.
- Dentist enters the password on the specific required field.
- If the required entered information is not correct and does not match that saved in the system database, then an error message is displayed.
- Dentist remains in the Log in page.
- Dentist should enter the required information again in order to log in.

Dentist – Profile Update

- Dentist is logged in the system.
- Dentist may make changes to his/her information by accessing the Profile Page on the Profile button at the side of his/her page.

In the Profile page, the following information is provided:

- Name – Surname – Birthday – Degree – Phone – Email – Gender – Password - Profile Photo – CV - Status
- Dentist may make changes to this information on the button Edit.
- Dentist may save the changes by the Submit button.

Dentist - Upcoming Appointment

- Dentist is logged in the system.
- To see the upcoming appointment, the dentist can access the page by the Button Check the upcoming appointment and the appointment with the specific information will be provided.

Dentist - See Schedule

- Dentist is logged in the system.
- To see the schedule directly, the dentist can access the page on the Check your schedule button and the schedule will be shown.

Dentist – See Appointments and Schedule

- Dentist is logged in the system.
- Dentist goes to Appointments page.

[E-DENT] Requirements Specification

- Dentist is provided with the following options:

Appointment Number – Date – Time – Service – Comment – Name - Surname- Special Health Conditions

Done/Not Done

- Dentist can write on the specific fields to make the desired and corresponding annotations.
- Dentist is provided with the Daily Schedule in this page.

Dentist – Complete and See Patient Dental Records

- Dentist is logged in the system.
- Dentist can access the Patient Dental Records page by the button with the same name.
- Dentist is provided with the following information about the patient:
Name – Surname – Age – Gender - Treatment and services provided to the patient
- Dentists completes all the patients' dental records.

Dentist – Message the Patients and received messages

- Dentist is logged in the system.
- Dentist is provided with the messages from the patients in the first page.
- Dentist can write a message by the Write a message button at the end of the page.

Dentist – Log out

- Dentist is logged in the system.
- Dentist can log out by the Log out button on the right of the page.
- Dentist is logged out and is directed to the Log in page.

Economist fails to log in:

- The log in form to be filled with the credentials input is shown
- Economist types either username or password incorrectly
- The alert is shown for the incorrect credentials pop-up
- The alert for failing to log in is shown
- Request to refill the credentials with the right input is requested

Economist logs in successfully:

- The log in form to input the credentials is shown
- Economist types the username and password correctly
- After the database check is done the economist is directed to his homepage

[E-DENT] Requirements Specification

- The economist views his personal homepage

Economist fails to change his credentials and profile info:

- In the upper-right of economists homepage clicks the Profile button
- The personal info “My Profile” page is shown showing all the credentials
- Clicks on the EDIT button on the bottom and all the fields can be edited
- Make the desired changes and click the SAVE button
- Changes not made according to validation rules and “Failed to save changes” pops-up

Economist changes his credentials and profile info:

- In the upper-right of economists homepage clicks the Profile button
- The personal info “My Profile” page is shown showing all the credentials
- Clicks on the EDIT button on the bottom and all the fields can be edited
- Make the desired changes and click the SAVE button
- Changes made according to validation rules and “Changes successfully saved” pops-up

Economist checks the revenue (inflows):

- On his homepage there is a small table with the header Incomes
- The very last transactions are shown there
- To open the full table clicks Go Button next to the table
- The table shows all the transactions:
number name surname date time description amount.

Economist Sorts the inflow transactions (all steps above):

- The economist can sort them by all of them by clicking on the table headers

Economist Chooses the extent of the transactions list (all steps above):

- The economist can expand the list by clicking on the “Show” dropdown button and choose:
 - Today
 - This Week
 - This Month
 - This Year

Economist Prints a list of transactions (all steps above):

- By clicking Print button, after having done the sorting and expansion
- The print Tab Pops-up

Economist Checks the Expenses:

On his homepage there is a small table with the header Incomes

1. The very last transactions are shown there
2. To open the full table clicks Go Button next to the table

Economist Adds an expense transaction (all steps above):

- Clicks on the “ADD” Button on the bottom right
 - A new editable row is shown in the beginning of the list

[E-DENT] Requirements Specification

- After finishing filling the fields on the row click button “Save”
- The table shows all the transactions with the headers:

number name surname date time description amount.

Economist Sorts the expense transactions (all steps above):

- The economist can sort them by all of them by clicking on the table headers

Economist Chooses the extent of the transactions list (all steps above):

- The economist can expand the list by clicking on the “Show” dropdown button and choose:
Today This Week This Month This Year

Economist Prints a list of expense transactions (all steps above):

- By clicking Print button, after having done the sorting and expansion
- The print Tab Pops-up

Economist controls the inventory depot:

- From his homepage on the upper-right clicks Inventories
- The Inventory page opens, a list of materials is shown each entry having:
Material Amount Duration Description Last Order Date Last order amount Price per Unit

Economist Adds a new material:

- Click button ADD
- An editable row of blank entries shows in the beginning of the list
- After filling the row, click the button Save
 - If entries filled according to the validation rules the row is created on the list
 - If entries are filled in violation of the validation rules, a warning will show asking to refill

Economist Deletes an existing material :

- Select the existing material by clicking its row
- Click the DELETE button, an alert pops-up “Do you really want to delete this material”
- Click Yes, the entry is removed from the list

Economist Edits an existing material:

- Select the existing material by clicking its row
- Then click the EDIT button
- The Selected row becomes editable, click and edit the entries
- To finish click SAVE
 - If entries filled according to the validation rules the row is updated
 - If entries are filled in violation of the validation rules, a warning will show asking to refill

Economist Orders new materials:

- When a material low on stock to reorder click “Go” button next to “Place new order:”
 - A list with the names of suppliers as well as their emails and phone numbers will show
 - By clicking their emails, automatically an email window will open where the economist can make the order

Economist Checks Employees Payroll

- On his homepage on the upper right selects Employee’s payroll:
- The payroll list page is loaded, the list of employees is shown each one having:
 - Name
 - Surname

[E-DENT] Requirements Specification

- c. Job
- d. Wage
- e. Last wage paid Date
- f. Wage paid (Yes/No)

Economist adds a new employee to the payroll:

- Click button ADD
- An editable row of blank entries shows in the beginning of the list
- After filling the row, click the button Save
 - 1. If entries filled according to the validation rules the row is created on the list
 - 2. If entries are filled in violation of the validation rules, a warning will show asking to refill (redo step 5)

Economist Deletes an employee:

- Select the employee row by clicking his/her row
- Click the DELETE button, an alert pops-up “Do you really want to delete this row”
- Click Yes, the entry is removed from the list

Economist Edits an employee wage information :

- Select the employee row by clicking it
- Then click the EDIT button
- The Selected row becomes editable, click and edit the entries
- To finish click SAVE
 - 1. If entries filled according to the validation rules the row is updated
 - 2. If entries are filled in violation of the validation rules, a warning will show asking to refill (return to 5)

Economist takes report request from admin:

- On the homepage “Document request” notification is shown
- By clicking on the notification the request specifications are shown
- After the economist has prepared the report clicks the Upload button on the notification
- Selects the report file
- Press Send Button

Patient Scenarios

Successful Log in:

- Patient enters the username on the specific required field correctly.
- Patient enters the password on the specific required field correctly.
- If this information is correct and the database check is done then the patient is successfully logged in.
- The patient is directed to his/her personal page where are provided the following options:
 - Patient's profile - Check dental clinic's services - Book an appointment
 - Check appointment status - My Appointments - Dental Records

-Chat with the dentist - Log out

Unsuccessful Log in:

- Patient enters the username on the specific required field.
- Patient enters the password on the specific required field.
- If this information is incorrect and does not match that saved in the database, then an error message is displayed.
- Patient remains in the Log in page.
- Patient must re-enter the required information in order to log in.

Patient - Update profile

- Patient is logged in the system.
- Patient clicks on the Profile button at the side of his/her page.

At the Profile page this information is provided:

-Name -Surname-Age-Birthday-Email-Phone-Gender-Special Health Condition-Password-Picture

- Patient may make changes to this information, by the Edit button.
- Patient may save the changes by the Submit button.

Patient - View the services provided by the dental clinic

- To check dental clinic's services the patient should click on the button Check our services.
- By clicking on that button the list of the services provided will be shown to the patient.

Patient - Book an appointment

- Patient is logged in the system.
- If the patient decides to have a treatment/consult in the clinic, he/she should book an appointment by entering the time, date and service that he/she would like and then click on the Book now button.

Patient - Check appointment status

- Patient is logged in the system.
- To check appointment status, the patient simply has to click on the You can check here your appointment status button and the status of the appointment will be provided.
Status options will be as following:
 - Active: Patient makes a request
 - Confirmed: Receptionist has confirmed it
 - Cancelled: Receptionist/Patient has cancelled it
 - Finished: Service has been provided.
 - Completed: Patient has paid.

[E-DENT] Requirements Specification

- Patient may also check the appointment status by clicking My Appointments and he/she will be directed to Appointments page.

Patient - Appointments

- Patient is logged in the system.
- Patient is provided with the following information:
 - Appointment number-Date-Time-Service-Status-Doctor-Paid/Not Paid
- Patient can make changes on the appointment by clicking on the button Edit.
- Patient can save the changes by clicking on the button Submit.
- In this page the patient will be provided the following information related to the status of the appointment:
 - Active: Patient makes a request
 - Confirmed: Receptionist has confirmed it
 - Cancelled: Receptionist/Patient has cancelled it
 - Finished: Service has been provided.
 - Completed: Patient has paid.

Patient - View Dental Records

- Patient is logged in the system.
- Patient clicks Records and it shows his/her dental records and also the beginning and the end date of the dental treatment.

Patient - Chat with the dentist

- Patient is logged in the system.
- If the patient wants to chat with the doctor, just click the button “chat with the doctor”, then communicate with the dentist.

Patient - Log out

- Patient is logged in the system.
- To log out, the patient should just click on the top right corner on “Log out” button.
- Patient is logged out and is sent back to the log in form page.

4.2 Use Cases

Name	UC_1.1	Log in
Summary	User enters the system by providing the required credentials.	
Actor	Admin, Receptionist, Dentist, Patient, Economist	
Description	User enters the password and username, and if match those saved in database, users are directed inside the system, where can perform different activities.	
Precondition	User must have a valid account to proceed with log in. The account is created by the admin.	
Alternative	There is no alternative option. Each user must have an account.	
Post Condition	User is directed to his/her page. User can access the system.	

Name	UC_1.2	Log out
Summary	Use the log out feature provided for every user	
Actor	All users	
Description	After the user is done using the system he can exit from his account via log out.	
Precondition	Users must be logged in.	
Alternative	There are no alternatives	
Post Condition	Exit from the account and redirect to the log in page	

[E-DENT] Requirements Specification

Name	UC_1.3	Change credentials/Edit profile
Summary		Users can change their account credentials or the information details in their own profile.
Actor		All users
Description		Users can change the username and password or other profile information according to the validation rules
Precondition		Users must edit only their own profiles and enter data complying with the predefined rules.
Alternative		The fields that can be changed are picked by the users and after replacing them with new information user may either save or cancel the changes
Post Condition		The updated profile is shown and if credentials are changed the user is redirected to the log in page.

Name	UC_2.1	View/edit the employees' list
Summary		The admin user can manage the employees adding, removing or just checking them in the list.
Actor		Admin
Description		The admin user when logged in can use the functionality of viewing the Employee list provided on the interface
Precondition		Admin user must be logged in.
Alternative		Adding deleting or just viewing, and after the required edit is performed the changes may be saved or canceled.

[E-DENT] Requirements Specification

Post Condition	Add/Delete/View employees on the list.
-----------------------	--

Name	UC_2.2	View/edit the list of services
Summary		The admin user can manage the list of the services provided by the clinic.
Actor		Admin
Description		The admin user can use the functionality shown in the interface to Add/Remove service or change the prices.
Precondition		Admin user must be logged in, enter the required information to add new service or change price.
Alternative		Add/Delete/View services on the list or change specific prices. the required edit is performed the changes may be saved or canceled.
Post Condition		Received info about the services and edited them if needed.

Name	UC_2.3	Request reports
Summary		Admin requests needed reports prepared by the economist and views the received ones.
Actor		Admin
Description		Admin requests financial or inventory reports by the economist which he can view after they are received and sent to the economist any message or comment.
Precondition		Admin must be logged in.

[E-DENT] Requirements Specification

Alternative	Admin can pick Balance sheet/Income Statement/Wages report/Cash-flow statement out of the financial reports or the inventory record report
Post Condition	The current reports are reviewed by the admin or the requests/comments are sent to economist.

Name	UC_2.4	Manage the job applications
Summary		The admin user can manage job application incoming from applicant through the application
Actor		Admin
Description		The admin can view, reject applications, invite for an interview or contact with the applicant
Precondition		Admin user must be logged in.
Alternative		The admin can only view the application or can reject/ask a interview.
Post Condition		Received info about the application and perform the actions if needed.

Name	UC_3.1	Apply for a job
Summary		The applicants applies for a job position posted on the E-Dents page
Actor		Receptionist
Description		Applicant fills the application form with the required information and submits it
Precondition		The user must fill the application according to the rules
Alternative		No alternative option.

[E-DENT] Requirements Specification

Post Condition	Submitted applications
----------------	------------------------

Name	UC_3.2.1	Create a patient account
Summary		User creates a patient account that is added to the system.
Actor		Receptionist,Patient
Description		Receptionist/patient creates a patient account by filling the form with required information. The patient can then access the system.
Precondition		The receptionist must fill all the required fields based on information provided by the patient.(or patient)
Alternative		Receptionists can create/delete patients' accounts.
Post Condition		The patient can now enter the system. The patient is added to the patient's list and shows in the receptionist and dentist page.

Name	UC_3.2.2	Delete a patient account
Summary		Receptionist deletes a patient account from the patient's list.
Actor		Receptionist
Description		Only the receptionist can delete a patient account that leaves the clinic. The patient account is removed from the database.
Precondition		The user must be logged in the system and must have a list of patients opened to complete the process.

[E-DENT] Requirements Specification

Alternative	Admin can also delete the user accounts.
Post Condition	The patient cannot access the system. The patient is not on the patient's list shown in the receptionist and dentist page.

Name	UC_3.2.3	Search a patient account
Summary		User searches a patient by username or id on the Patient page.
Actor		Receptionist, Dentist
Description		The user enters username or id related to the patient. After the search is pressed the Patient page displays the patient he/she is looking for.
Precondition		The user must be logged in the system. A valid username or id must be entered in the search field.
Alternative		No alternative option.
Post Condition		The user is displayed the searched patient page.

Name	UC_4.1	View the services provided by the dental clinic
Summary		User view the services that are provided.
Actor		Patient
Description		For checking dental clinic's services patient should search for the button Check our services.
Precondition		The user must be logged in the system.
Alternative		After finding that button the list of services provided will be shown.
Post Condition		User can accept and add them in the service lists.

[E-DENT] Requirements Specification

Name	UC_4.2.1	Book an appointment
Summary		The user can book an appointment online with his/her account.
Actor		Patient
Description		If the patient decides to have a treatment/consult in the clinic, he/she should book an appointment by entering the time, date and service that he/she would like and then search on the Book now button.
Precondition		The user must be logged in the system.
Alternative		No alternative option.
Post Condition		These appointments are added in Patients accounts.

Name	UC_4.2.2	Check appointment status
Summary		User can check the appointments online.
Actor		Patient
Description		To check appointment status, the patient simply has to find the You can check here your appointment status button and the status of the appointment will be provided. Status options will be as following: - Active: Patient makes a request - Confirmed: Receptionist has confirmed it - Cancelled: Receptionist/Patient has cancelled it - Finished: Service has been provided. - Completed: Patient has paid.
Precondition		User is logged in the system.
Alternative		There is no alternative option.
Post Condition		Patient may also check the appointment status by searching My Appointments button and he/she will be directed to Appointments page.

[E-DENT] Requirements Specification

Name UC_5.1.1	Add Appointment
Summary	The user adds new appointments for patients and dentists.
Actor	Receptionist
Description	The user enters the needed information on Appointment form such as patient name, dentist, service, date, time.
Precondition	The user must be logged in the system and be allowed to assign appointments.
Alternative	No alternative option.
Post Condition	New appointment is added to the Appointment and Schedules list.

Name UC_5.1.2	Check Appointment Request
Summary	User is notified for any appointment request from the patients.
Actor	Receptionist
Description	Receptionist receives notification of the requests from patients for appointment in his account.
Precondition	User must be logged in. User must have the notification option enabled.
Alternative	User can check if there is any request in the Appointment and Schedules page.
Post Condition	User can accept the request by adding it to the appointment list.

Name UC_5.1.3	Cancel Appointment
Summary	User cancel appointment for patients and dentists
Actor	Receptionist

[E-DENT] Requirements Specification

Description	The receptionist can cancel appointments scheduled from the appointment list by editing the Appointment list..
Precondition	The user must be logged in the system and be allowed to Edit and cancel appointments.
Alternative	Dentists can also cancel the appointment by emailing the patient.
Post Condition	The appointment is removed from the Appointment table in the database and removed from the list.

Name UC_5.2	View dentist schedule
Summary	User views list of appointments and schedules.
Actor	Receptionist, Dentist
Description	The user can view the dentist's schedules on the Appointment and Schedule page.
Precondition	The user must be logged in the system. At least one appointment should be scheduled and posted in the system.
Alternative	Dentist can view his/her appointment from his profile.
Post Condition	No post condition

Name UC_6.1	Upcoming Appointment
Summary	User views the upcoming appointment.
Actor	Dentist
Description	User can see the upcoming appointment on the main page and on the Appointments and Schedule page.

[E-DENT] Requirements Specification

Precondition	The user must be logged in the system. At least one appointment should pop up in the system.
Alternative	User can see the upcoming appointment on the Appointments and Schedule page.
Post Condition	No post condition.

Name UC_6.2	Add Patient Dental Records
Summary	User can create the patients' dental records.
Actor	Dentist
Description	User can make this action on the Patient Dental Records page.
Precondition	User must be logged in the system.
Alternative	No alternative option.
Post Condition	User must save changes after adding the Dental Records.

Name UC_6.	Edit/Remove Patient Dental Records
Summary	User can edit or remove the patients' dental records.
Actor	Dentist
Description	User can make this action on the Patient Dental Records page.
Precondition	User must be logged in the system.
Alternative	Dentist can delete or update the dental records.
Post Condition	User must save changes after performing the edit actions the Dental Records.

[E-DENT] Requirements Specification

Name UC_6.3	Message and communicate with the Patient
Summary	User chats with the patients
Actor	Dentist, Patient
Description	User communicates with the patients, receives and send messages on the main page where new messages are shown and the Write a message option is provided.
Precondition	User must be logged in the system.
Alternative	No alternative option.
Post Condition	No post condition.

Name UC_6.4	Remove Item
Summary	User views the item List.
Actor	Dentist
Description	User can access item list and remove the used item specifying name and quantity.
Precondition	The user must be logged in the system.
Alternative	There are no alternatives
Post Condition	Updated inventory.

Name UC_7.1.1	Full Inflow transactions
Summary	The user views the list of incoming transactions
Actor	Economist
Description	The economist accesses the list of all transactions from the clinic activity with their specifications in another page
Precondition	User must be logged in to his homepage

[E-DENT] Requirements Specification

Alternative	No alternative option.
Post Condition	The full page of inflow transactions (revenue) is shown

Name UC_7.1.2	Sorting of the revenue list
Summary	User sorts the transactions according to his preferences
Actor	Economist
Description	<p>The economist can sort the inflow transactions in relation with any of the properties of the transactions number</p> <ul style="list-style-type: none"> ● name ● surname ● date ● time ● description ● amount
Precondition	Economist must be loaded to the Revenue (transactions) page
Alternative	Economist can sort from upper or lower of the transaction properties
Post Condition	The list is sorted according to the requested property

Name UC_7.1.3	Filter Transaction
Summary	User chooses the extent of transactions shown on the list
Actor	Economist
Description	The economist can extend or constrict the list of transactions shown regarding the time frames they occurred.
Precondition	Economist must be loaded to the Revenue (transactions) page

[E-DENT] Requirements Specification

Alternative	Economist can choose between: <ul style="list-style-type: none"> ● Today ● This Week ● This Month ● This Year
Post Condition	The list updated to the desired time frame extent.

Name	Transactions printed
Summary	User prints the transaction
Actor	Economist
Description	The economist can print the list of transactions as desired.
Precondition	Economist must be loaded to the Revenue (transactions) page and have done the sorting and expansion.
Alternative	No alternative available
Post Condition	The print tab is loaded.

Name	Full Expense Transactions
Summary	The user views the list of expense transactions
Actor	Economist
Description	The economist accesses the list of all expenses incurred from the clinic activity with their specifications in another page
Precondition	User must be logged in to his homepage
Alternative	No alternative option.
Post Condition	The full page of expense transactions is shown

[E-DENT] Requirements Specification

Name	UC_7.2.2	Sorting of the expenses list
Summary	User sorts the transactions according to his preferences	
Actor	Economist	
Description	The economist can sort the expenses in relation with any of the properties of the transactions: <ul style="list-style-type: none"> ● number ● name ● surname ● date ● time ● description ● amount 	
Precondition	Economist must be loaded to the Expenses page	
Alternative	Economist can sort from upper or lower of the transaction properties	
Post Condition	The list is sorted according to the requested property	

Name	UC_7.2.3	Filter Expense Transactions
Summary	User chooses the extent of expense transactions shown on the list	
Actor	Economist	
Description	The economist can extend or constrict the list of transactions shown regarding the time frames they occurred.	
Precondition	Economist must be loaded to the Expenses page	
Alternative	Economist can choose between: <ul style="list-style-type: none"> ● Today ● This Week ● This Month ● This Year 	
Post Condition	The list updated to the desired time frame extent.	

[E-DENT] Requirements Specification

Name	UC_7.2.4	Expenses Transactions printed
Summary	User prints the expenses list	
Actor	Economist	
Description	The economist can print the list of Expenses as desired.	
Precondition	Economist must be loaded to the Expenses page and have done the sorting and expansion.	
Alternative	No alternative available	
Post Condition	The print tab is loaded.	

Name	UC_7.2.5	Add Expense
Summary	User adds an expense transaction	
Actor	Economist	
Description	The economist can add a new expense transaction by filling its required properties information	
Precondition	Economist must be loaded to the Revenue (transactions) page..	
Alternative	No alternative available	
Post Condition	The list is updated with the new expense transaction.	

Name	UC_7.3.1	Full Inventory Depot List
Summary	The user views the list materials in the inventory.	

[E-DENT] Requirements Specification

Actor	Economist
Description	The economist accesses the list of all materials on-hand for the clinic activity with their specifications in another page
Precondition	User must be logged in to his homepage
Alternative	No alternative option.
Post Condition	The full page of inventory is shown

Name	UC_7.3.2	Sorting of the materials list
Summary		User sorts the materials according to his preferences
Actor		Economist
Description		<p>The economist can sort the materials list in relation with any of the properties of them:</p> <ul style="list-style-type: none"> ● Material ● Amount ● Duration ● Description ● Last Order Date ● Last order amount ● Price per Unit
Precondition		Economist must be loaded to the Inventory page
Alternative		Economist can sort from upper or lower of the material properties
Post Condition		The list is sorted according to the requested property

Name	UC_7.3.4	Materials list printed
Summary		User prints the materials list
Actor		Economist
Description		The economist can print the list of materials as desired.

[E-DENT] Requirements Specification

Precondition	Economist must be loaded to the materials page and have done the sorting and expansion.
Alternative	No alternative available
Post Condition	The print tab is loaded.

Name	UC_7.3.5	Add Material
Summary	User adds a new material	
Actor	Economist	
Description	The economist can add a new material by filling its required properties information	
Precondition	Economist must be loaded to the Inventory page.	
Alternative	No alternative available	
Post Condition	The list is updated with the new material row..	

Name	UC_7.3.6	Select a Material
Summary	User selects a material	
Actor	Economist	
Description	The economist can select a material from the materials list to perform actions to it.	
Precondition	Economist must be loaded to the Inventory page.	
Alternative	Delete or Edit	
Post Condition	The list is updated without the deleted material/ with the edited material.	

[E-DENT] Requirements Specification

Name	UC_7.3.7	Order Material
Summary	User orders new materials.	
Actor	Economist	
Description	The economist can order new materials supplies through ways of communication with the suppliers.	
Precondition	Economist must be loaded to the Inventory page.	
Alternative	Email or Phone number of a certain supplier	
Post Condition	A pending order is shown.	

Name	UC_7.4.1	Full Employee Payroll List
Summary	The user views the Employees Payroll List.	
Actor	Economist	
Description	The economist accesses the list of all employees on the payroll with their wage specifications in another page	
Precondition	User must be logged in to his homepage	
Alternative	No alternative option.	
Post Condition	The full page of Payroll is shown	

Name	UC_7.4.2	Sort of Employees Payroll list
Summary	User sorts the employees according to his preferences	
Actor	Economist	
Description	The economist can sort the employees list in relation with any of the properties of them:	<ul style="list-style-type: none"> • Name • Surname

[E-DENT] Requirements Specification

	<ul style="list-style-type: none"> ● Job ● Wage ● Last wage paid Date ● Wage paid (Yes/No)
Precondition	Economist must be loaded to the Employees payroll page
Alternative	Economist can sort from upper or lower of the employees properties
Post Condition	The list is sorted according to the requested property

Name	UC_7.4.3	Print Employees payroll list
Summary	User prints the employees payroll list	
Actor	Economist	
Description	The economist can print the list employees payroll as desired.	
Precondition	Economist must be loaded to the Employees Payroll page and have done the sorting.	
Alternative	No alternative available	
Post Condition	The print tab is loaded.	

Name	UC_7.4.4	Add Employee
Summary	User adds a new employee on the payroll	
Actor	Economist	
Description	The economist can add a new Employee on the payroll by filling its required properties information	
Precondition	Economist must be loaded to the Employee Payroll page.	
Alternative	No alternative available	

[E-DENT] Requirements Specification

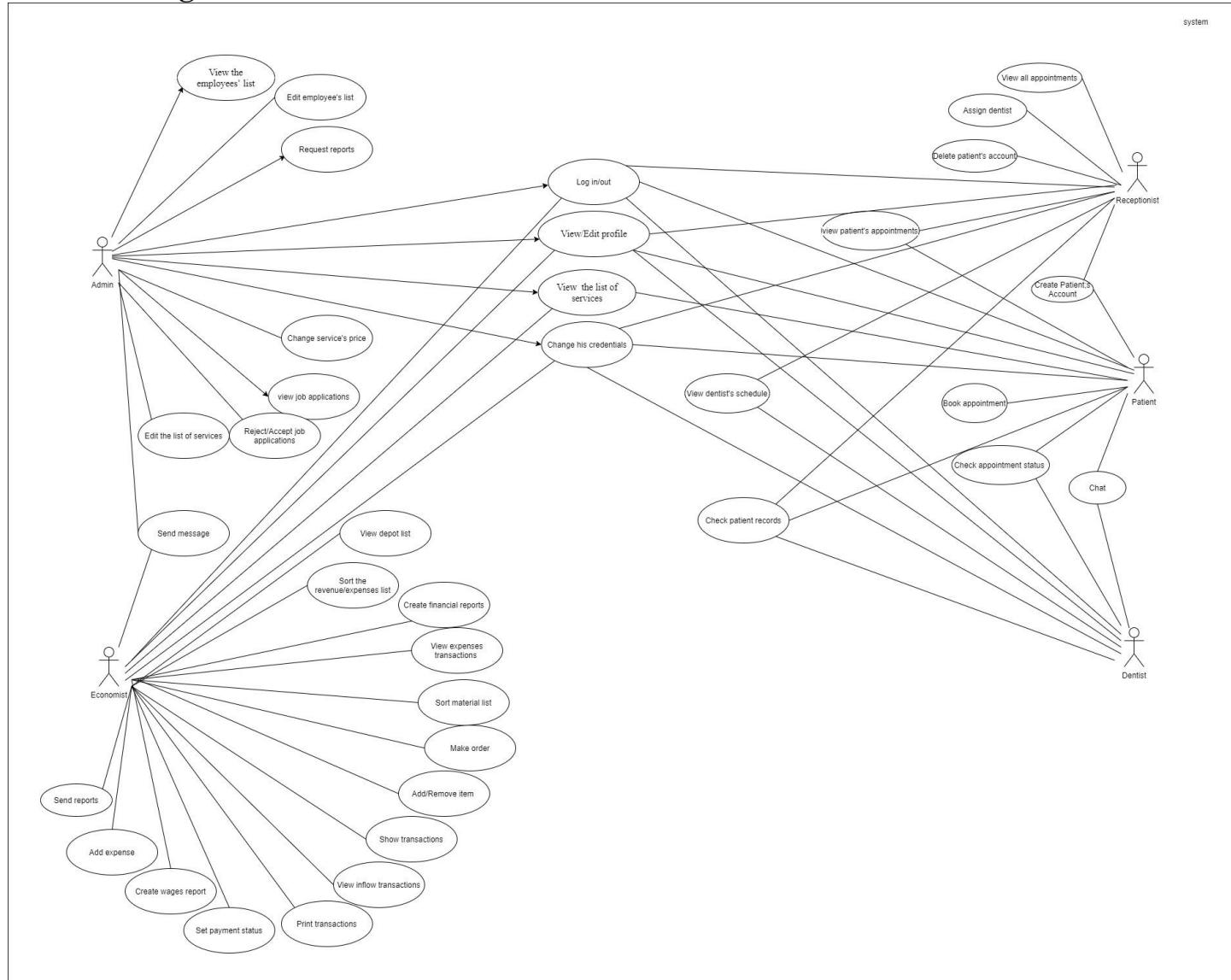
Post Condition	The list is updated with the new employee row.
----------------	--

Name UC_7.4.5	Selecting an Employee
Summary	User selects an Employee from the payroll
Actor	Economist
Description	The economist can select an employee from the payroll list to perform actions to it.
Precondition	Economist must be loaded to the Employees Payroll page.
Alternative	Delete or Edit
Post Condition	The list is updated without the deleted Employee/ with the edited Employee.

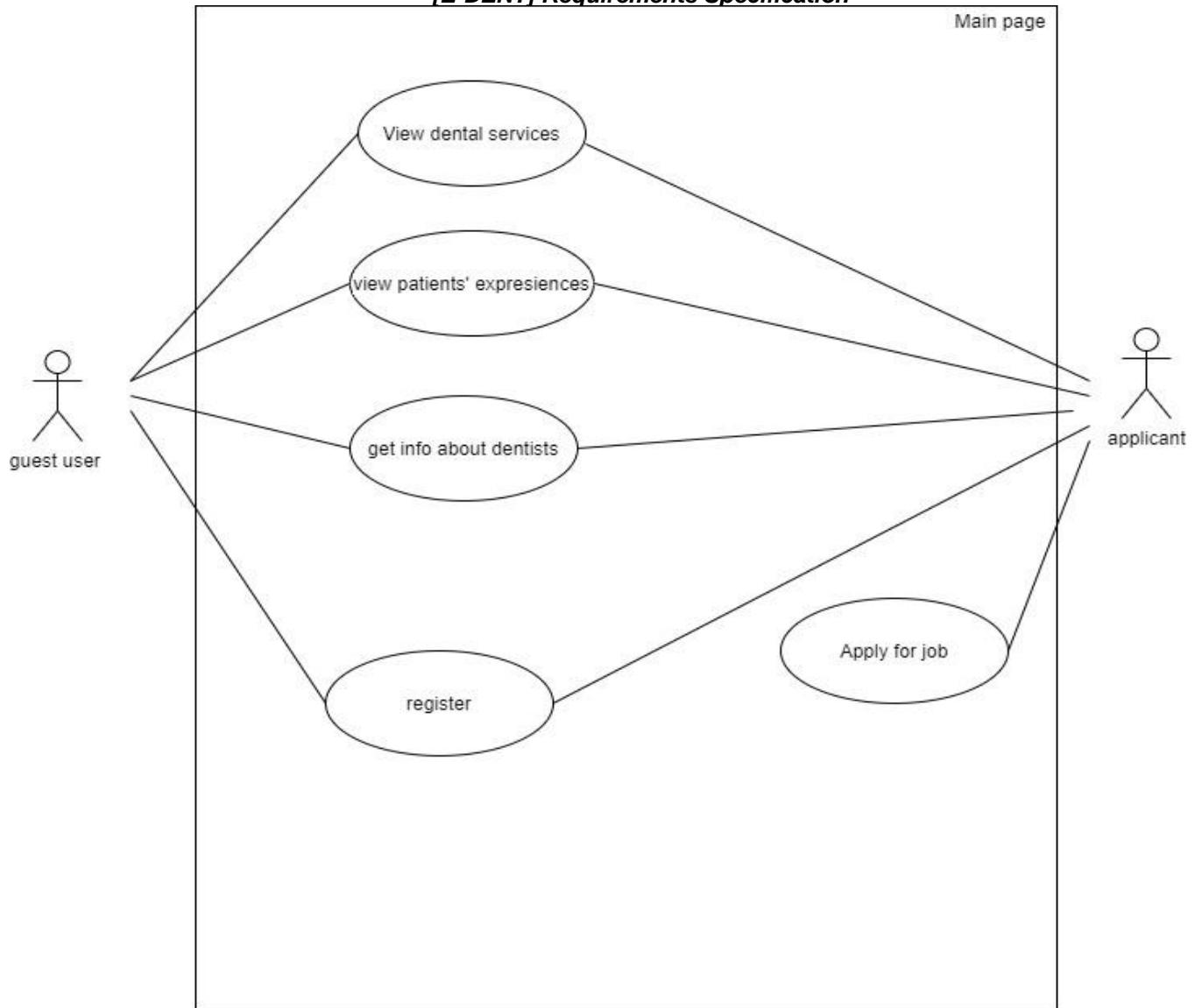
Name UC_7.5	Report Request
Summary	User takes a report request from the Admin.
Actor	Economist
Description	The economist is requested a financial/managerial report from the admin with specific requirements as stated from him.
Precondition	Economist must be loaded to his homepage.
Alternative	Upload, Reply
Post Condition	The file is uploaded and sent to admin/ the message is sent to admin.

4.3 Diagrams

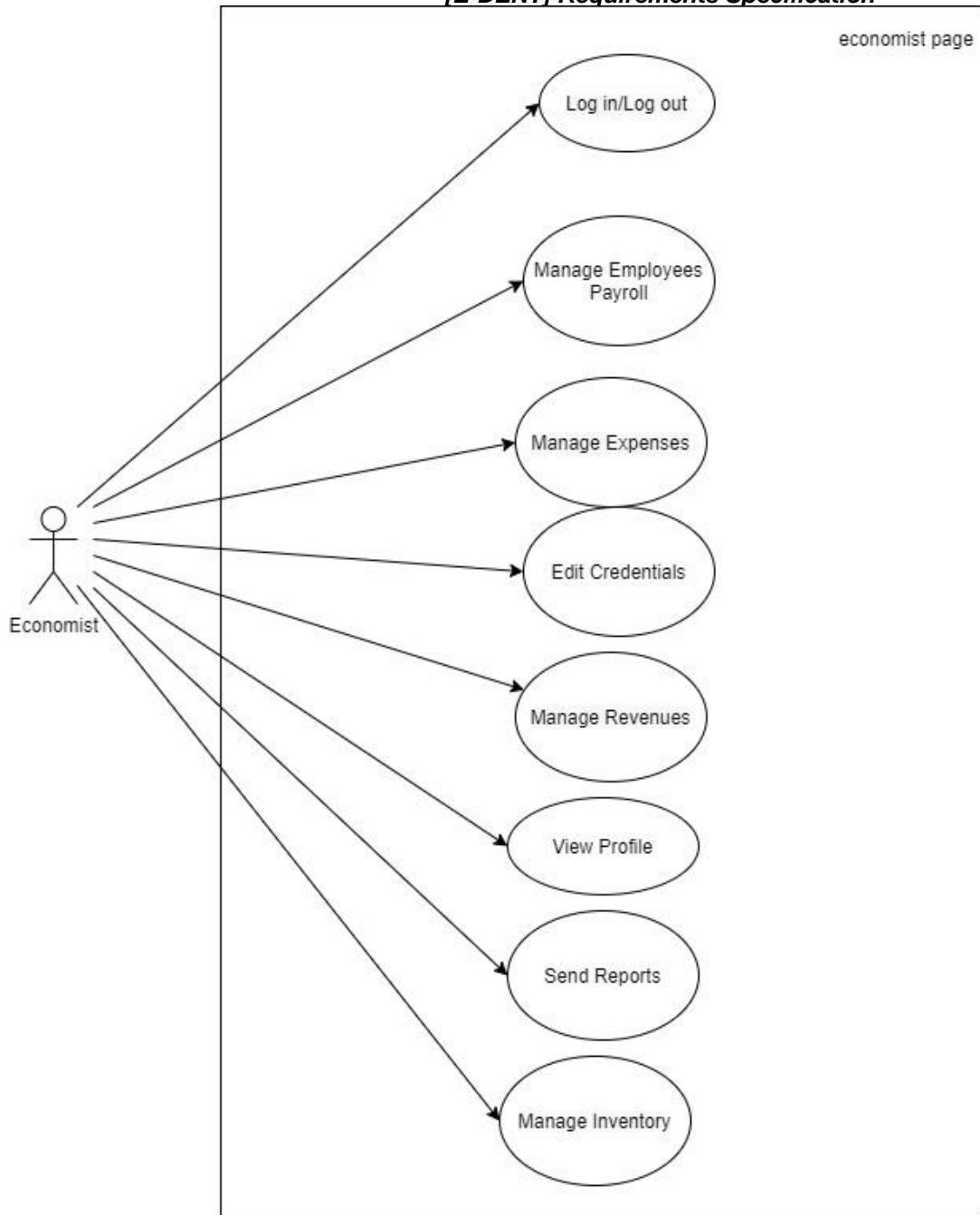
Use case Diagrams



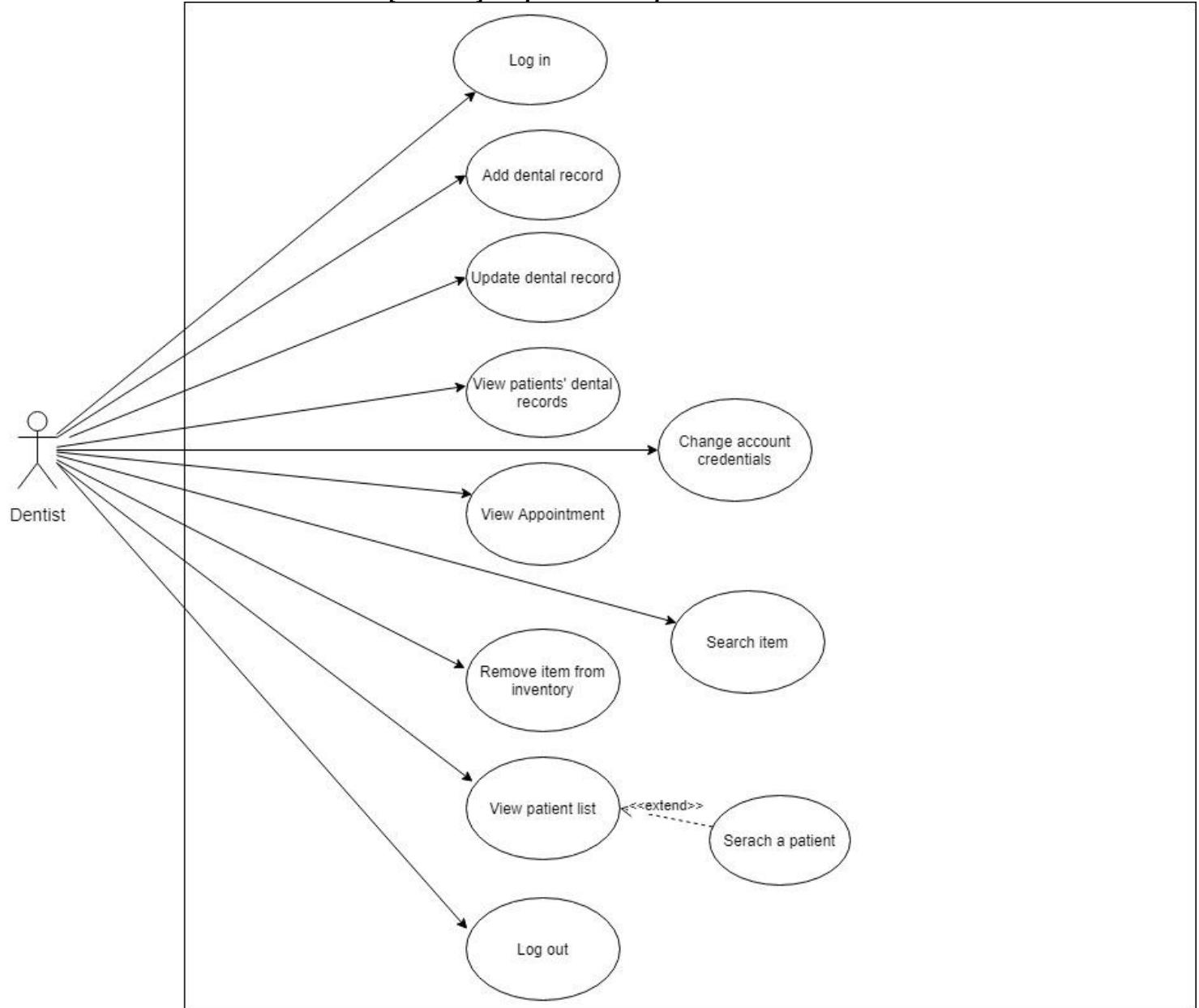
[E-DENT] Requirements Specification



[E-DENT] Requirements Specification



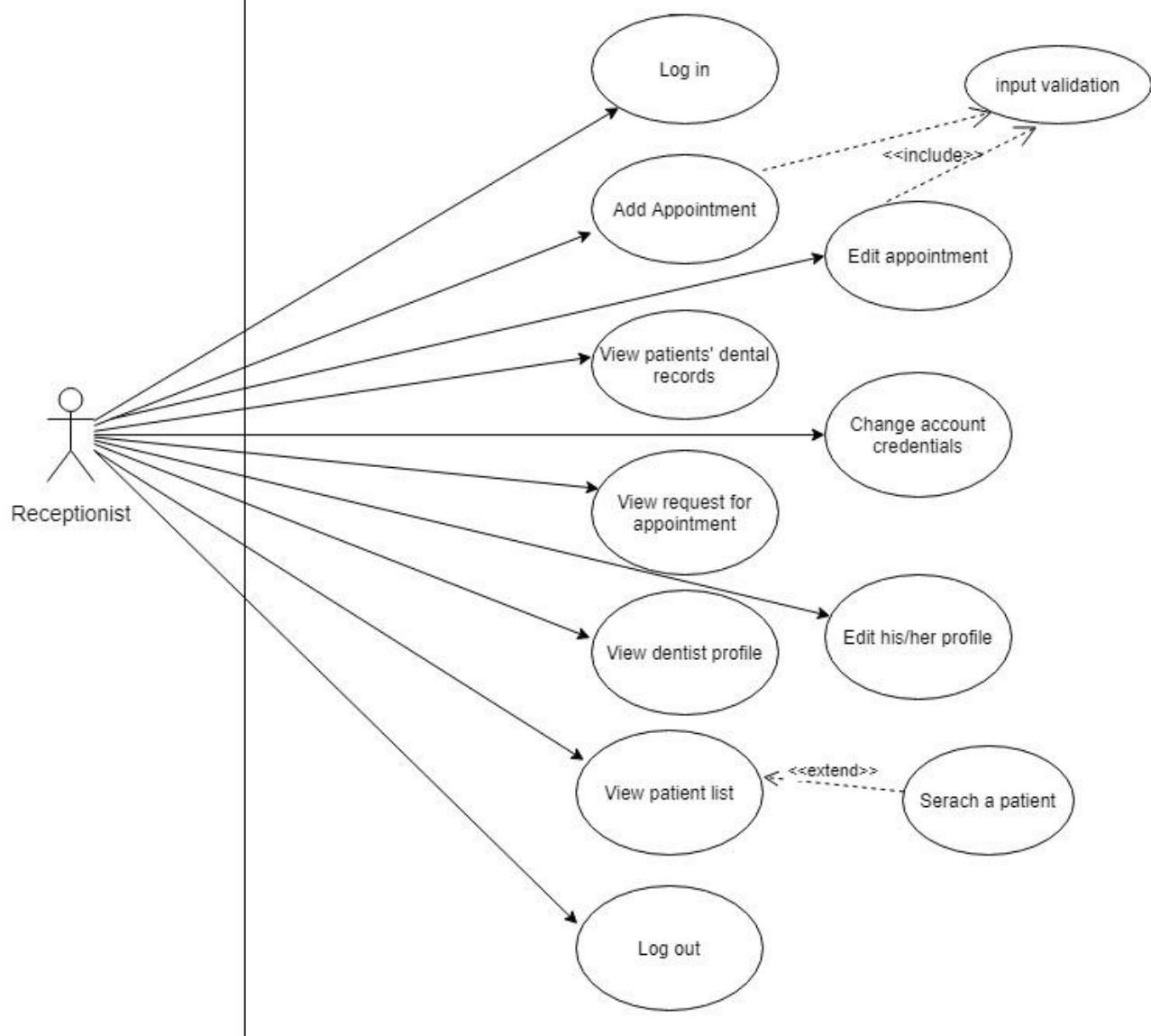
[E-DENT] Requirements Specification



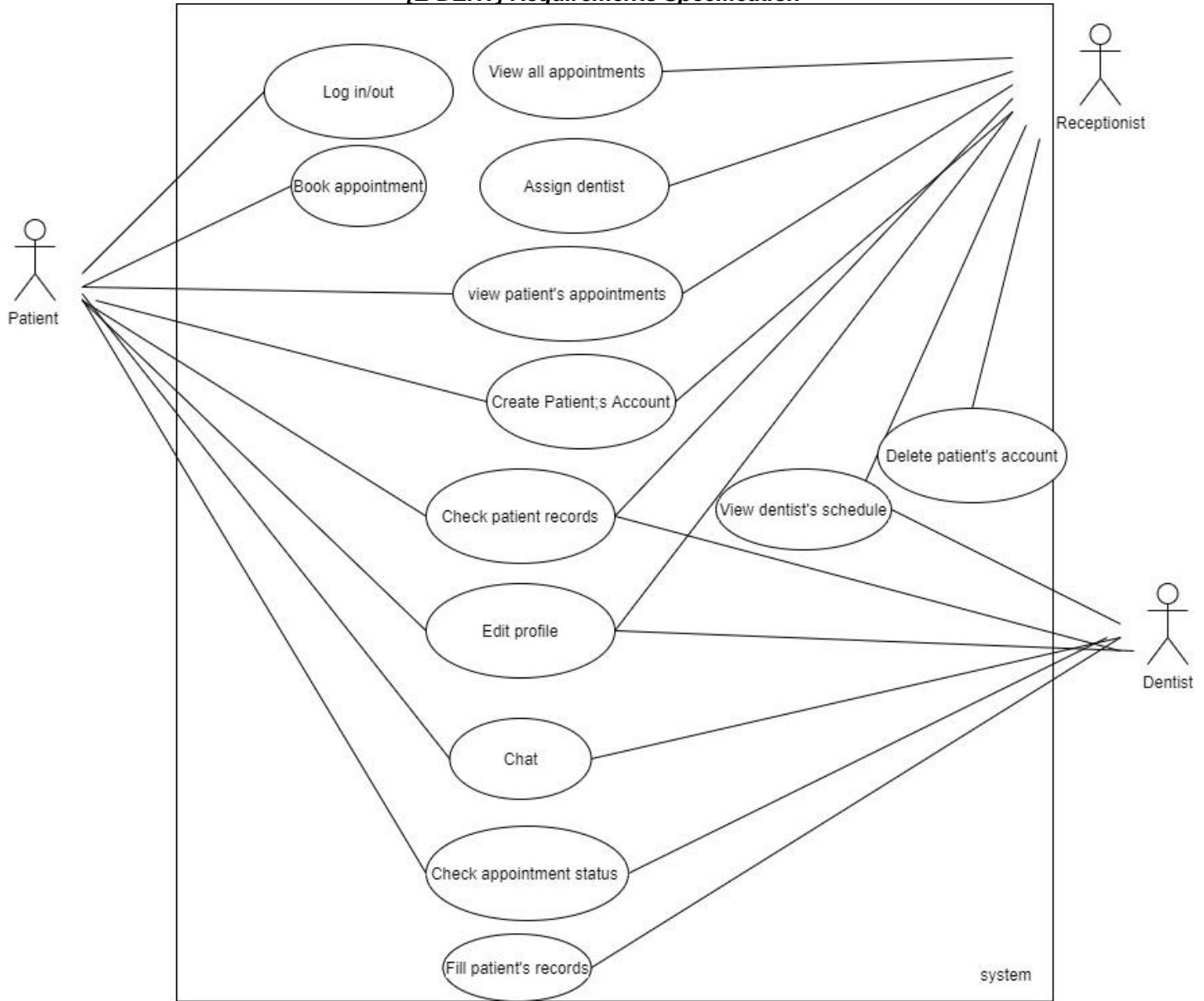
[E-DENT] Requirements Specification



[E-DENT] Requirements Specification

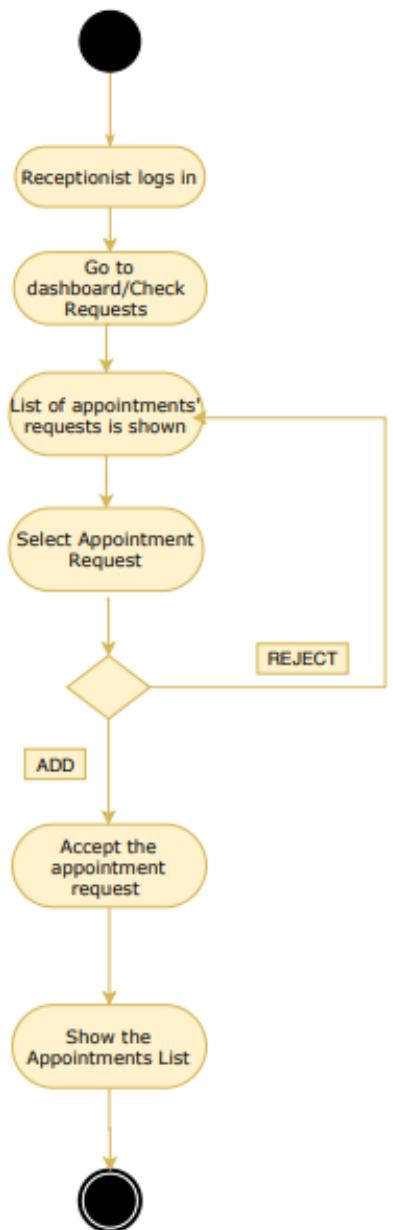


[E-DENT] Requirements Specification

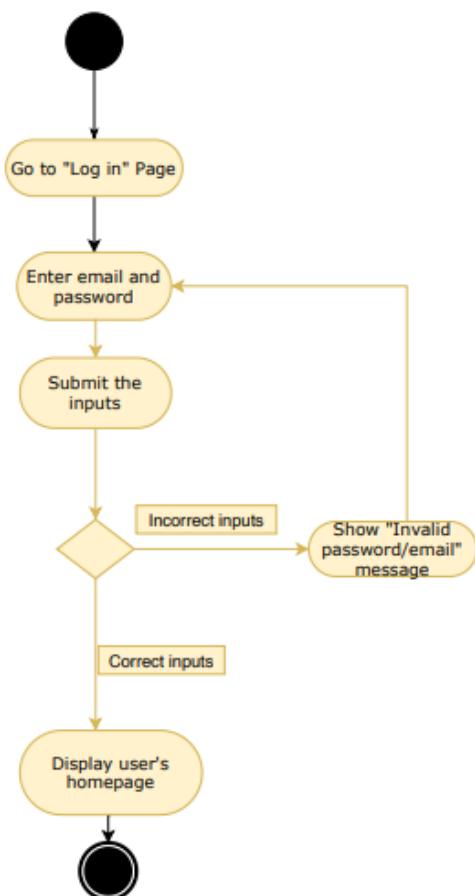


Activity diagrams

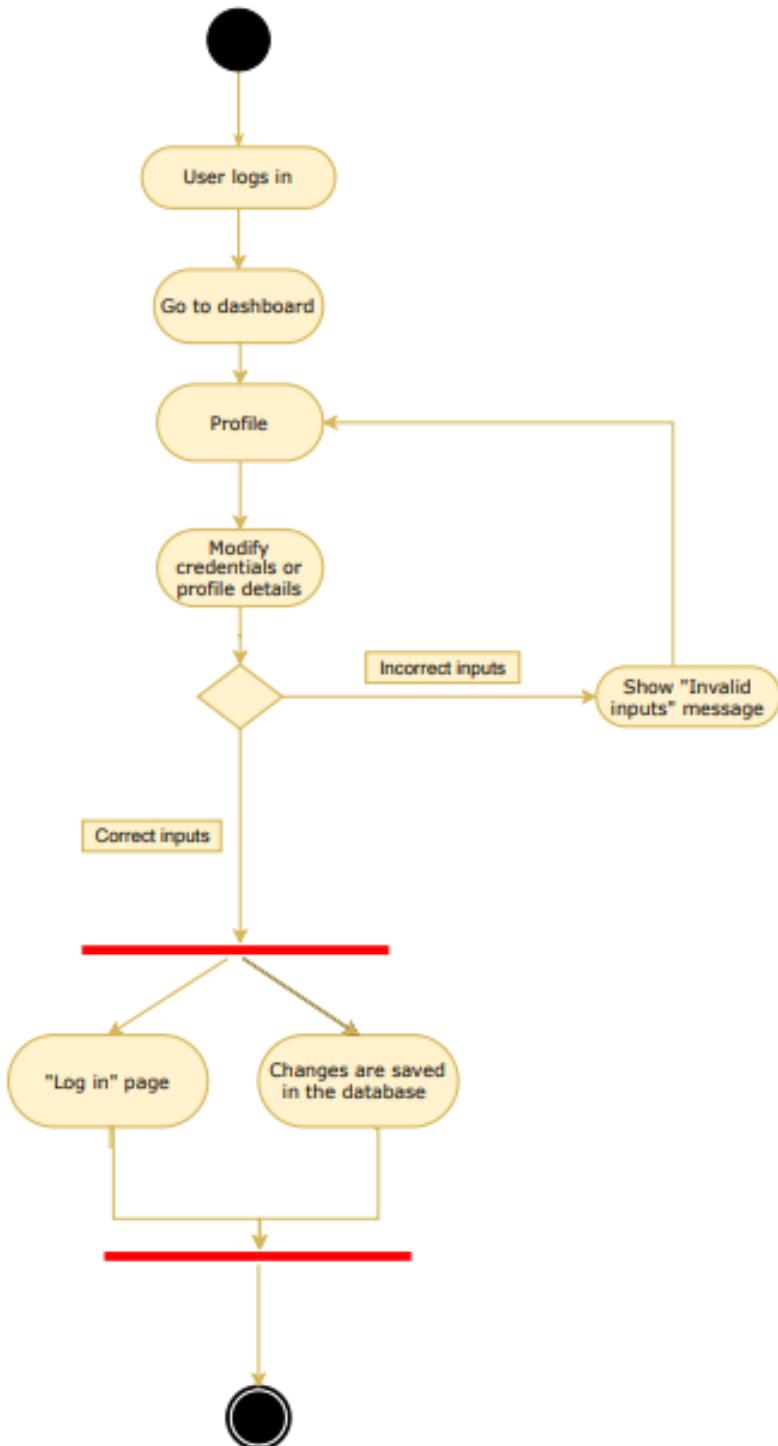
AC_05.1.2_CheckAppointmentRequest



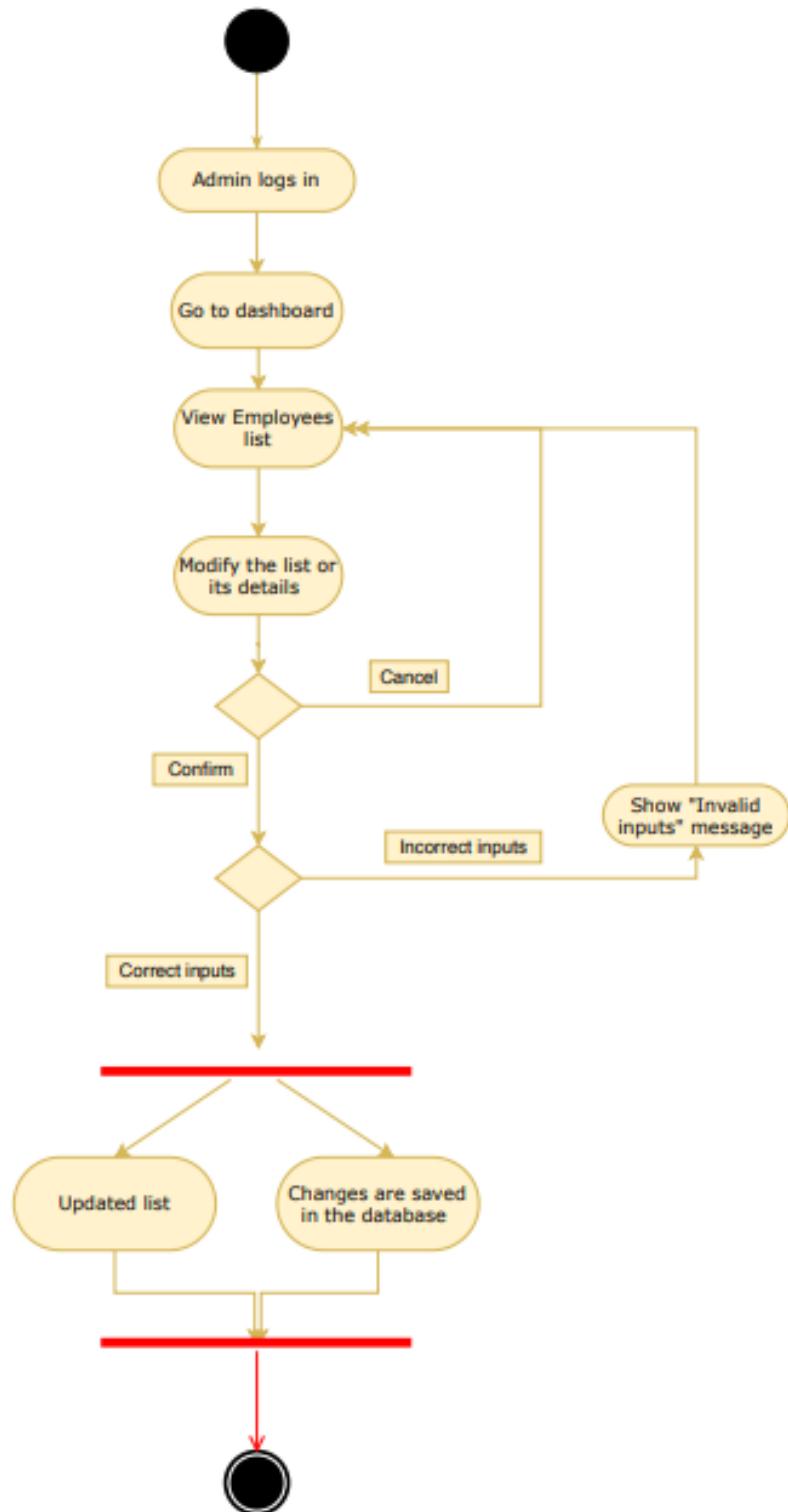
AC_01.1_Login



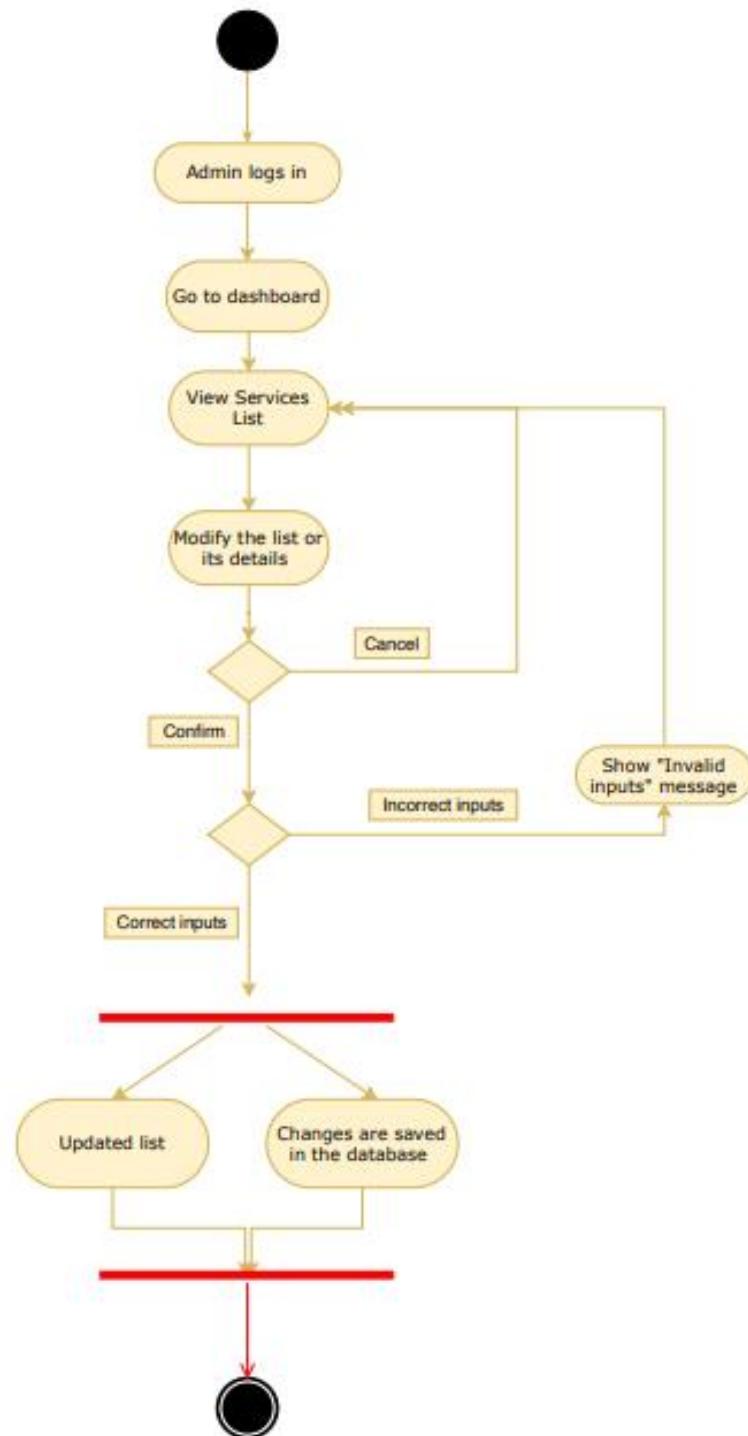
AC_01.3_EditProfile



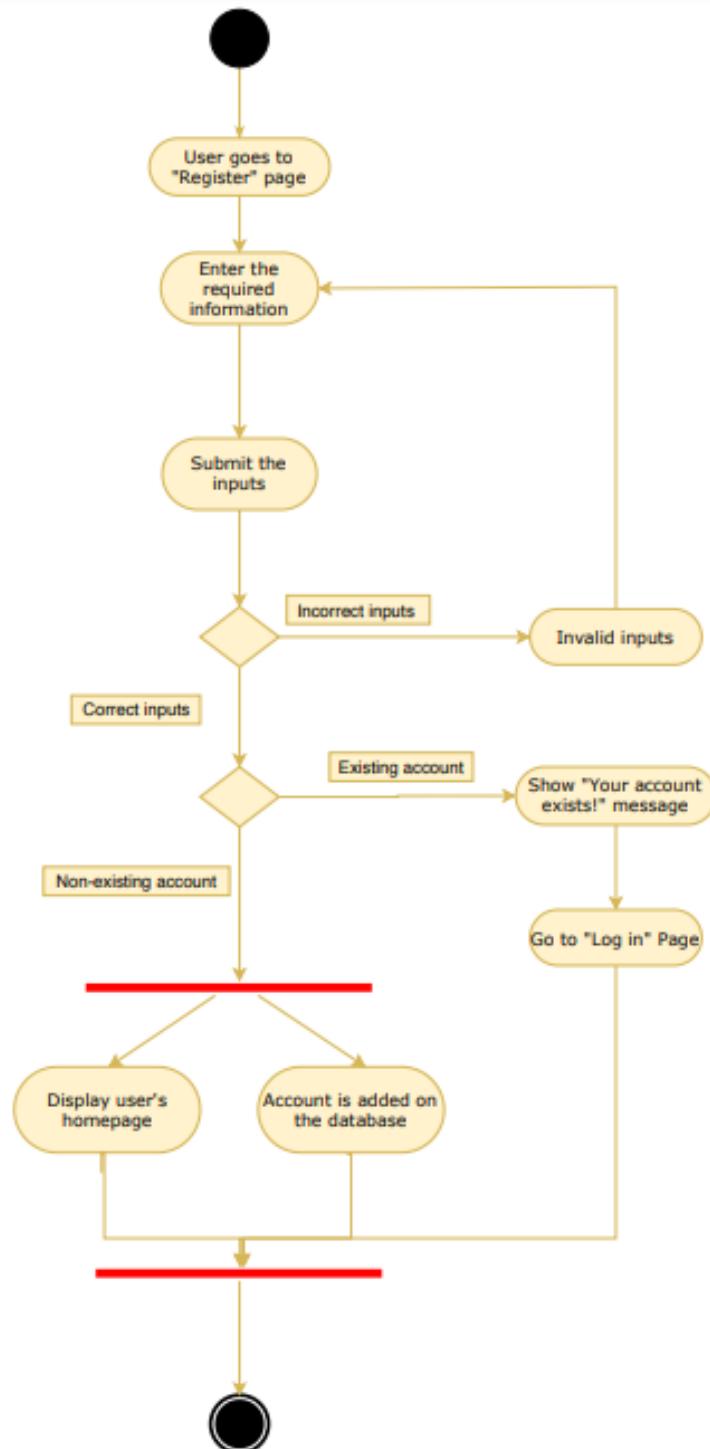
AC_02.1_ViewEditEmpList



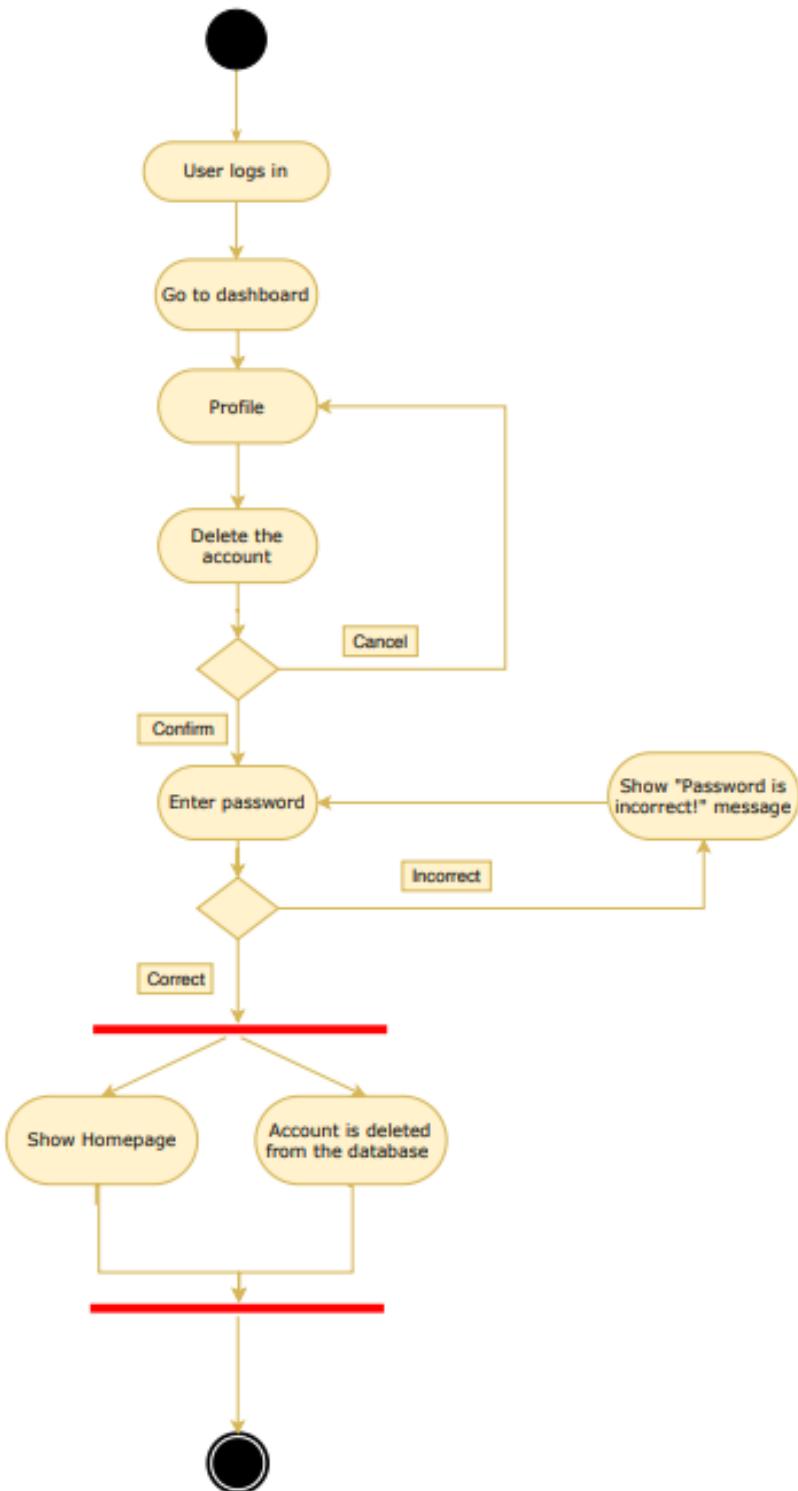
AC_02.2_ViewEditServiceList



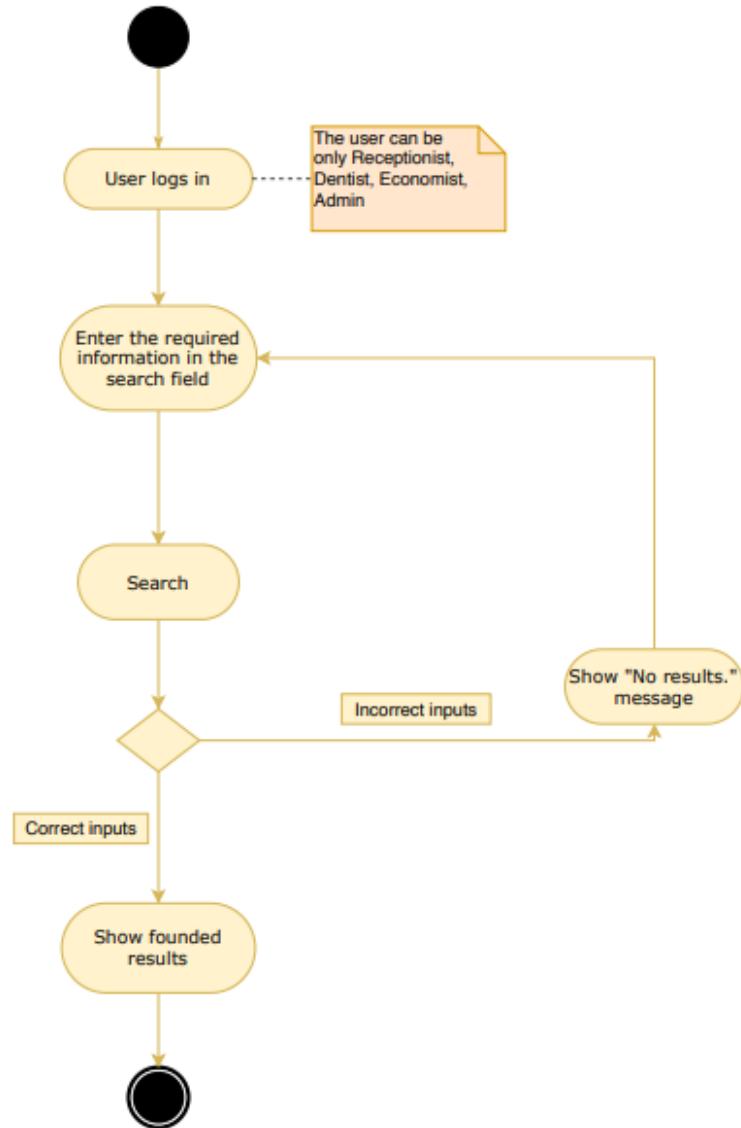
AC_03.2.1_CreateAccount



AC_03.2.2_DeleteAccount



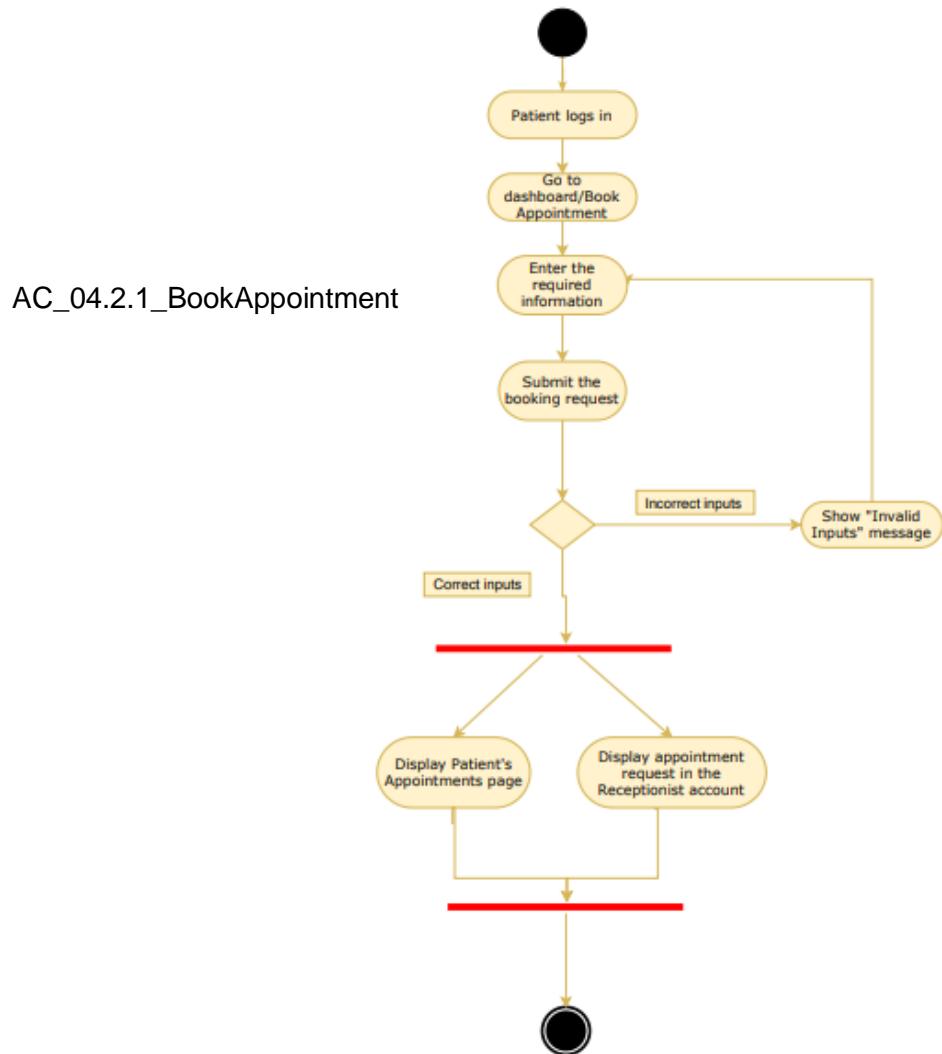
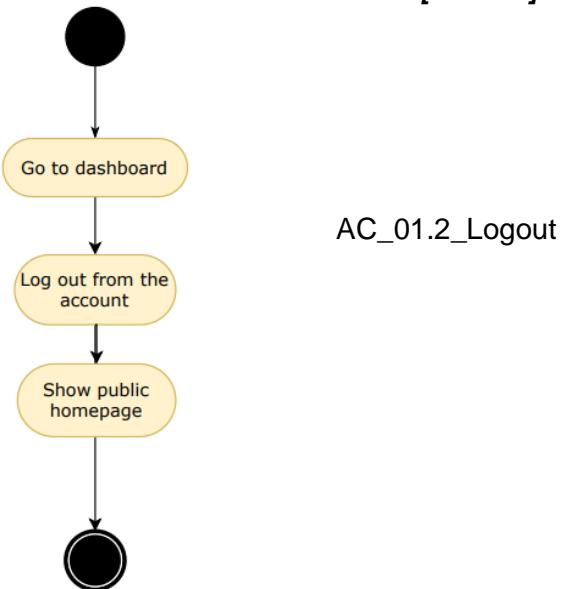
AC_03.2.3_SearchAccount



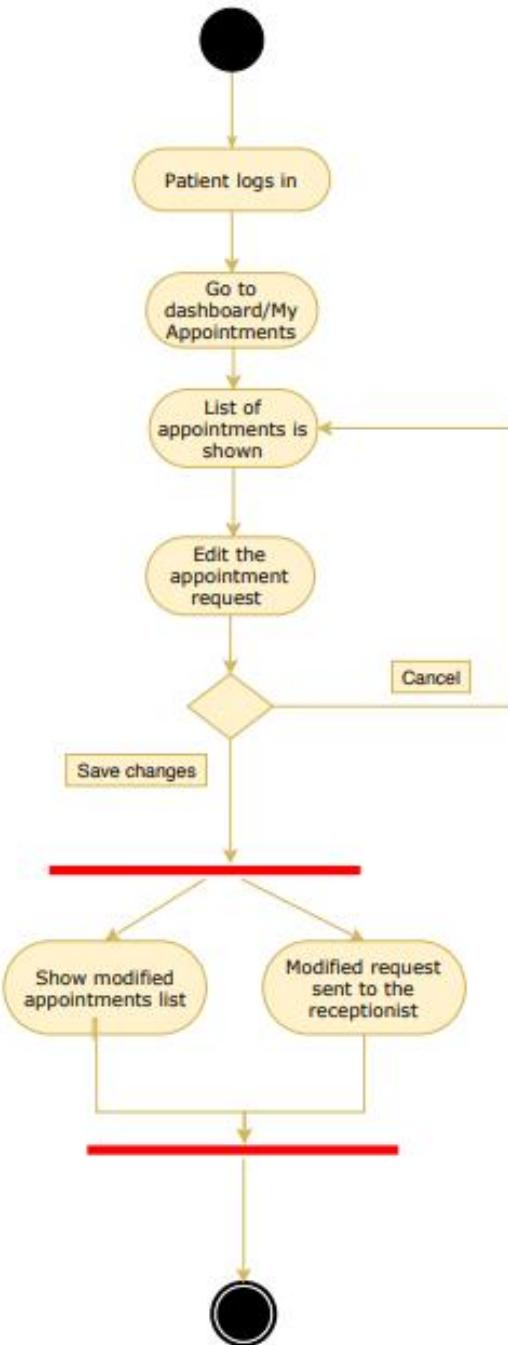
AC_04.1_ViewService



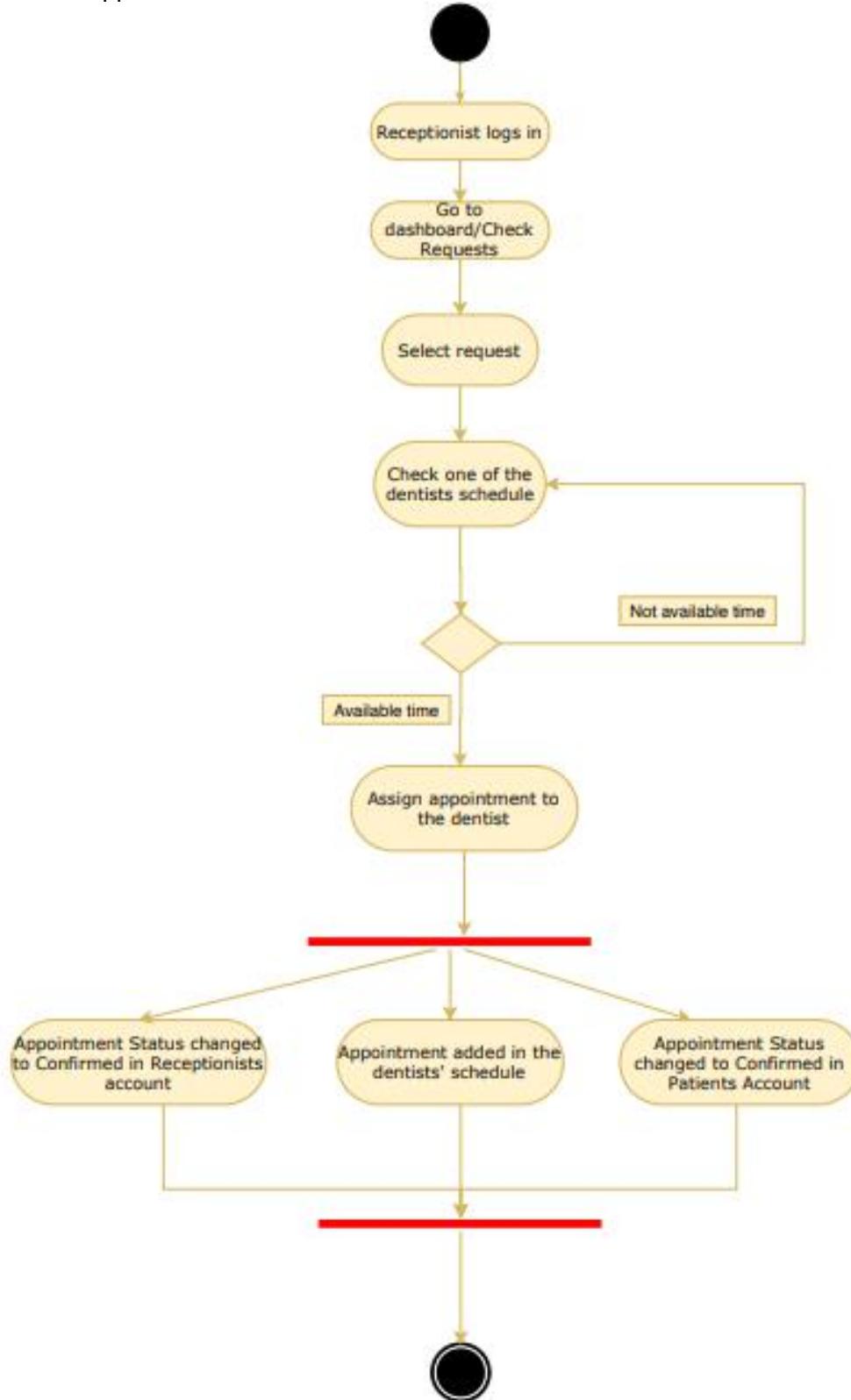
[E-DENT] Requirements Specification



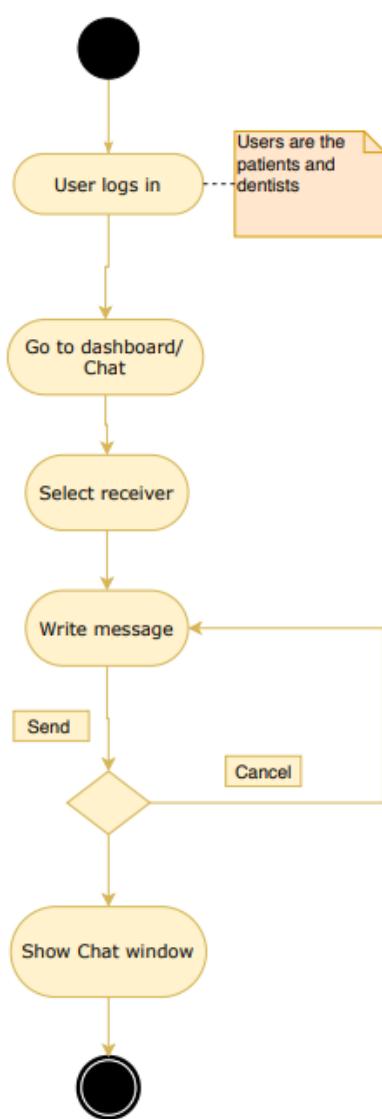
AC_04.2.2_CheckAppointment



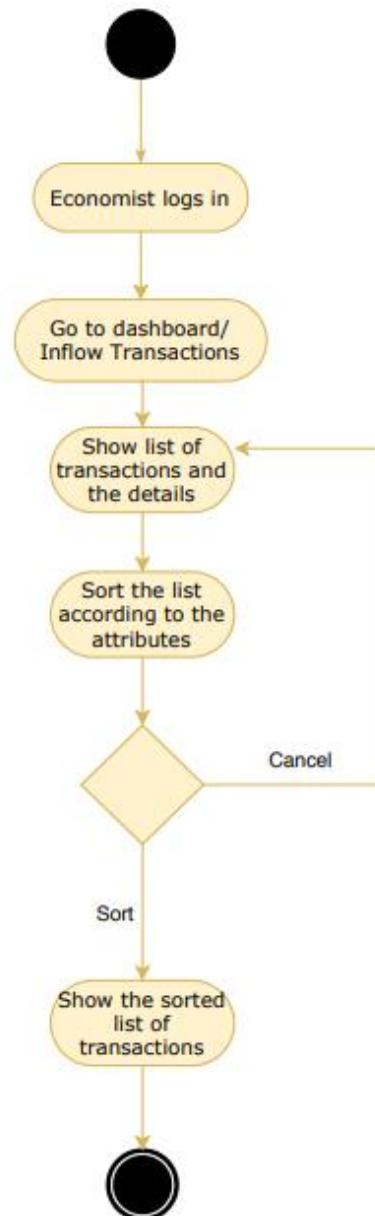
AC_05.1.1_AddAppointmen



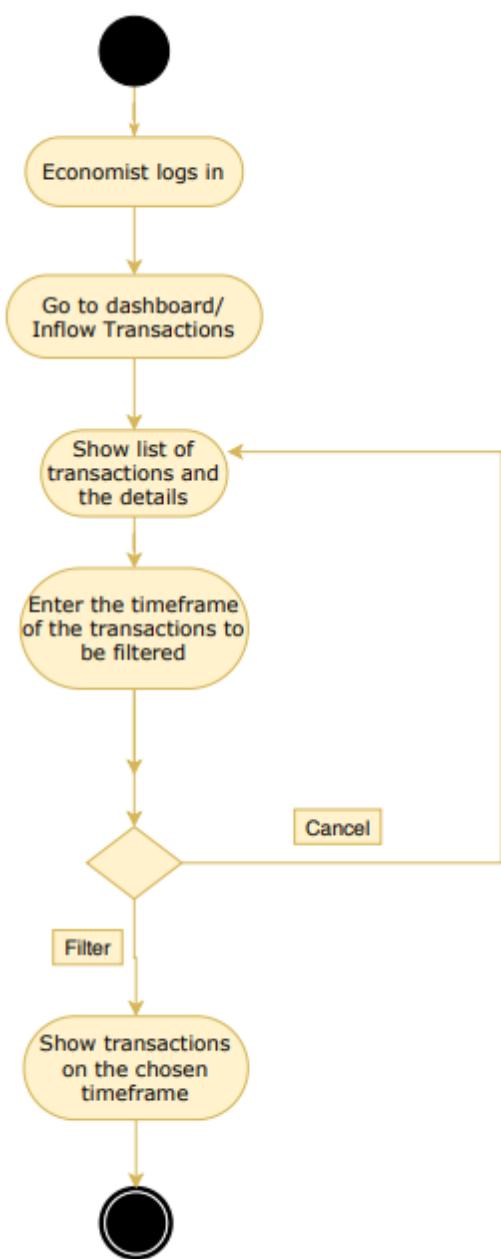
AC_06.3_Chat



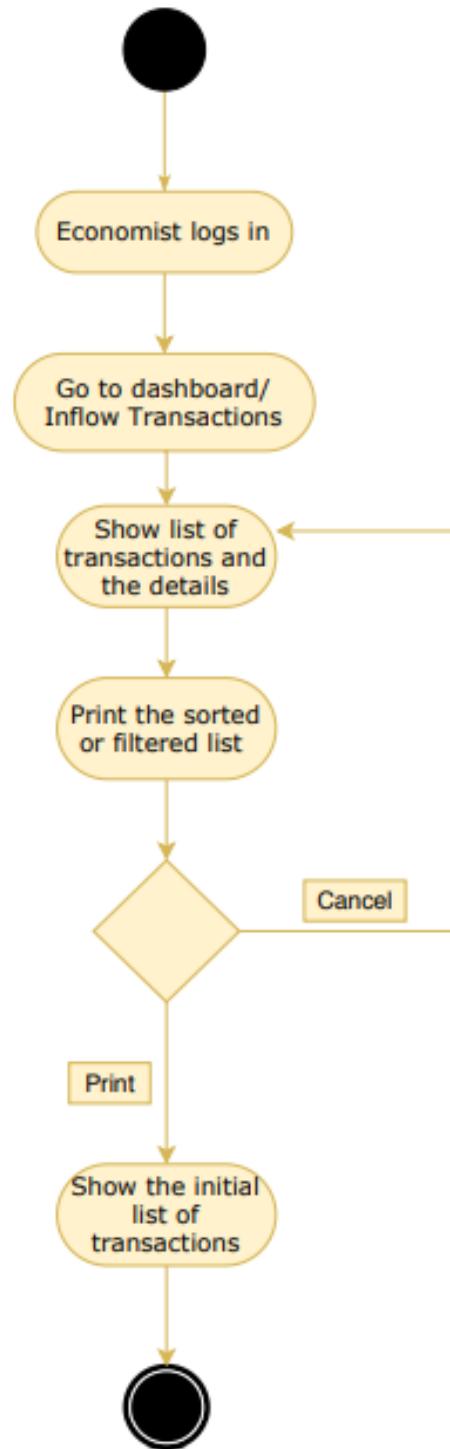
AC_07.1.1-2_InflowTransactions



AC_07.1.3_FilterTransaction

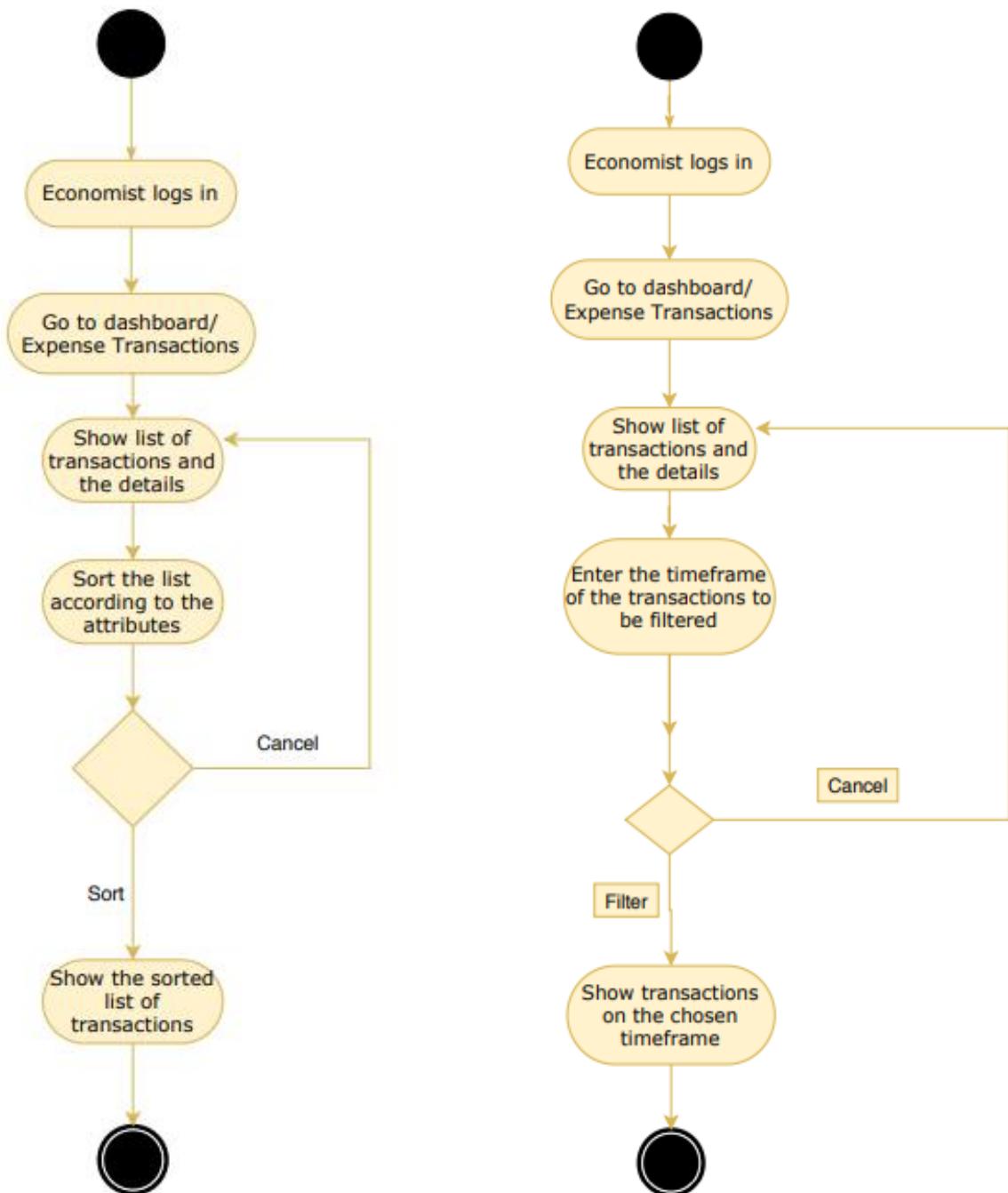


AC_07.1.4_PrintTransactions



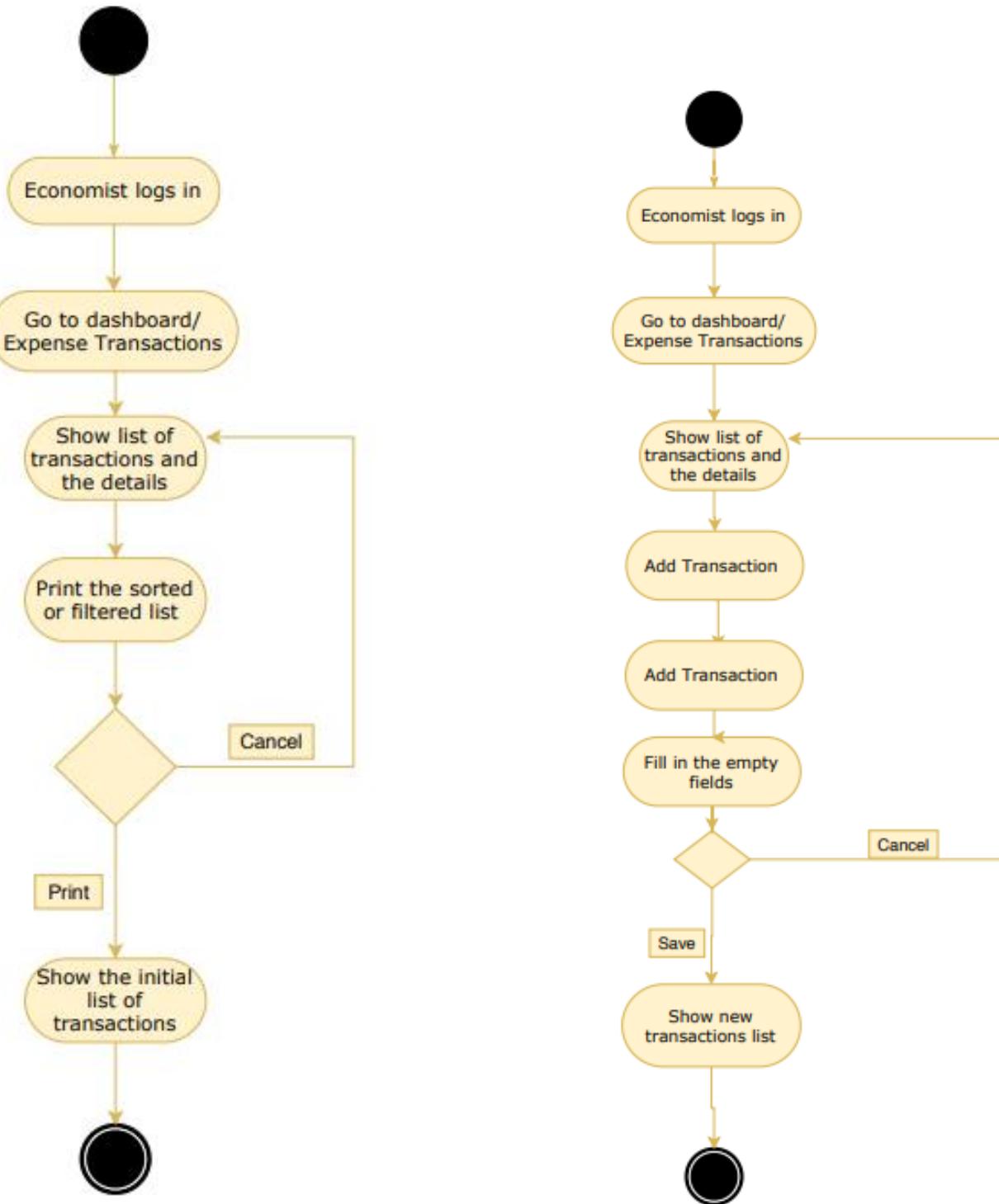
AC_07.2.1-2_ExpenseTransactions

AC_07.2.3_FilterExpenseTransaction



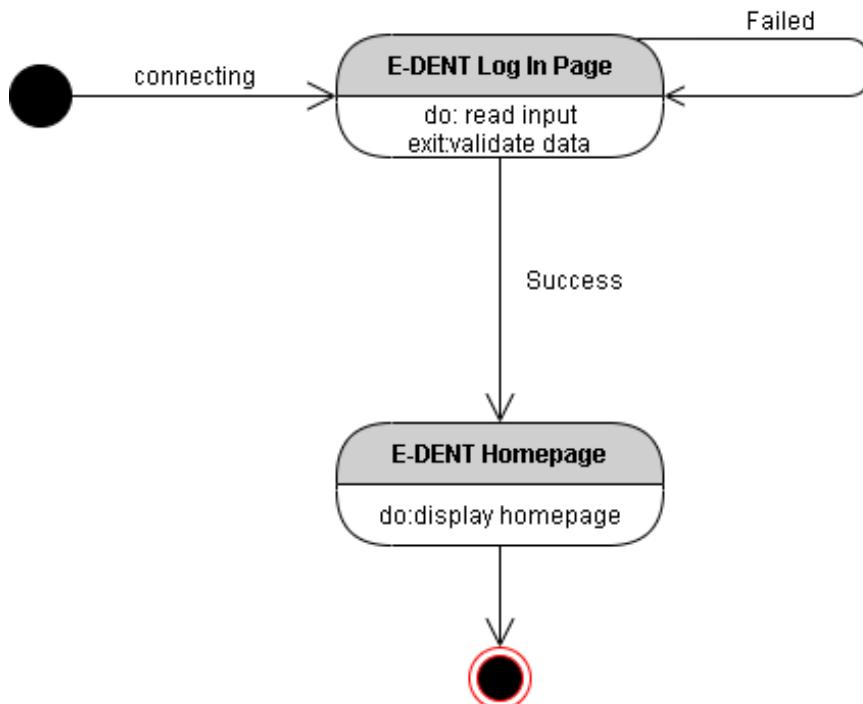
AC_07.2.4_PrintExpenseTransactions

AC_07.2.5_AddExpenseTransaction

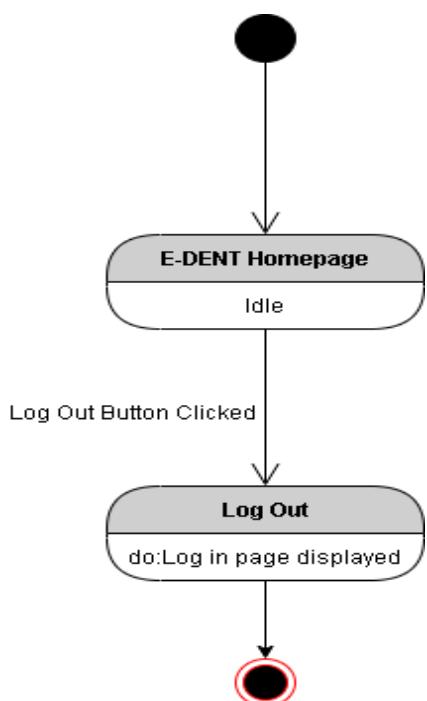


State Diagrams

ST_01.1_LogIn

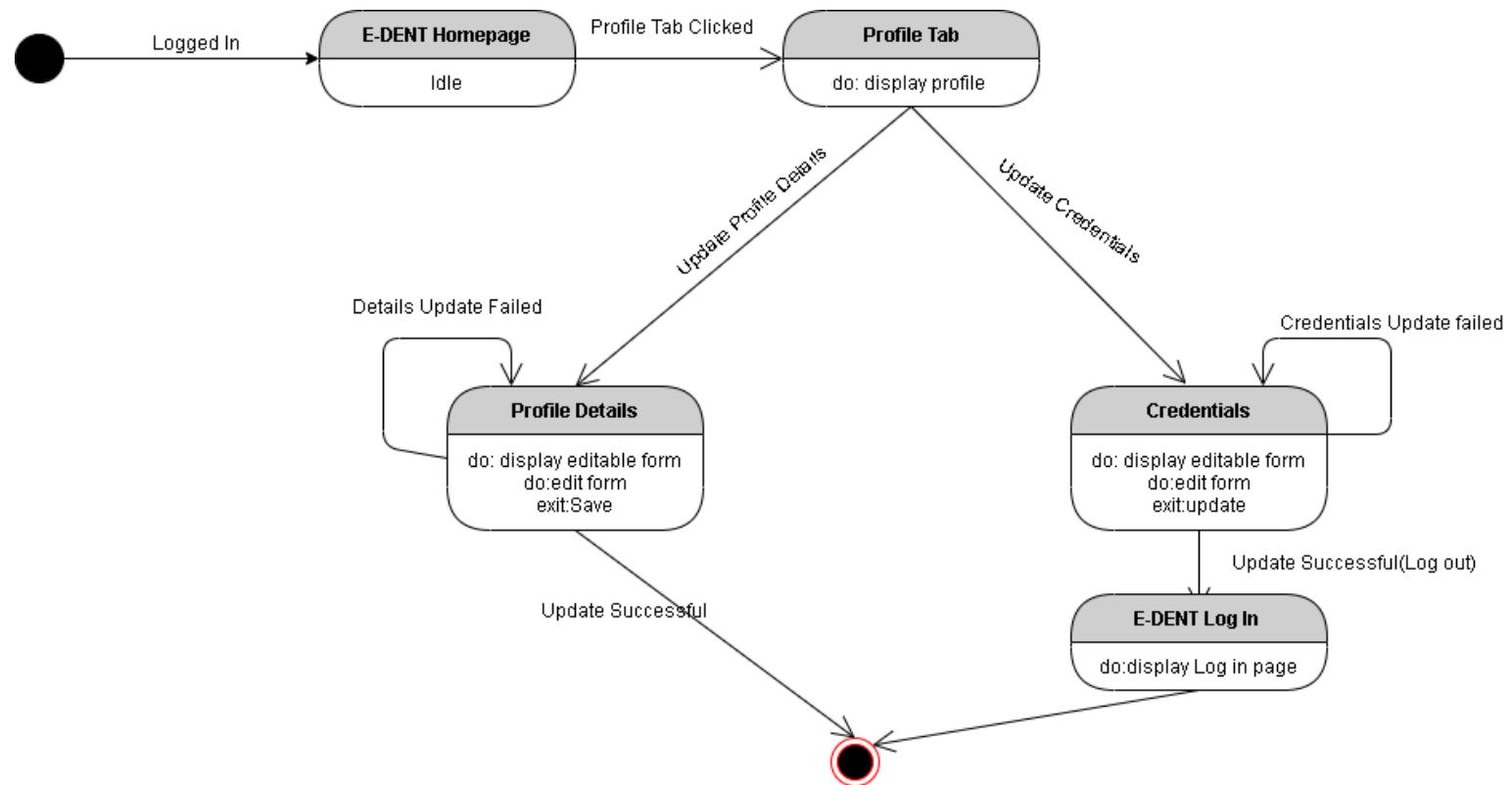


1.2_LogOut

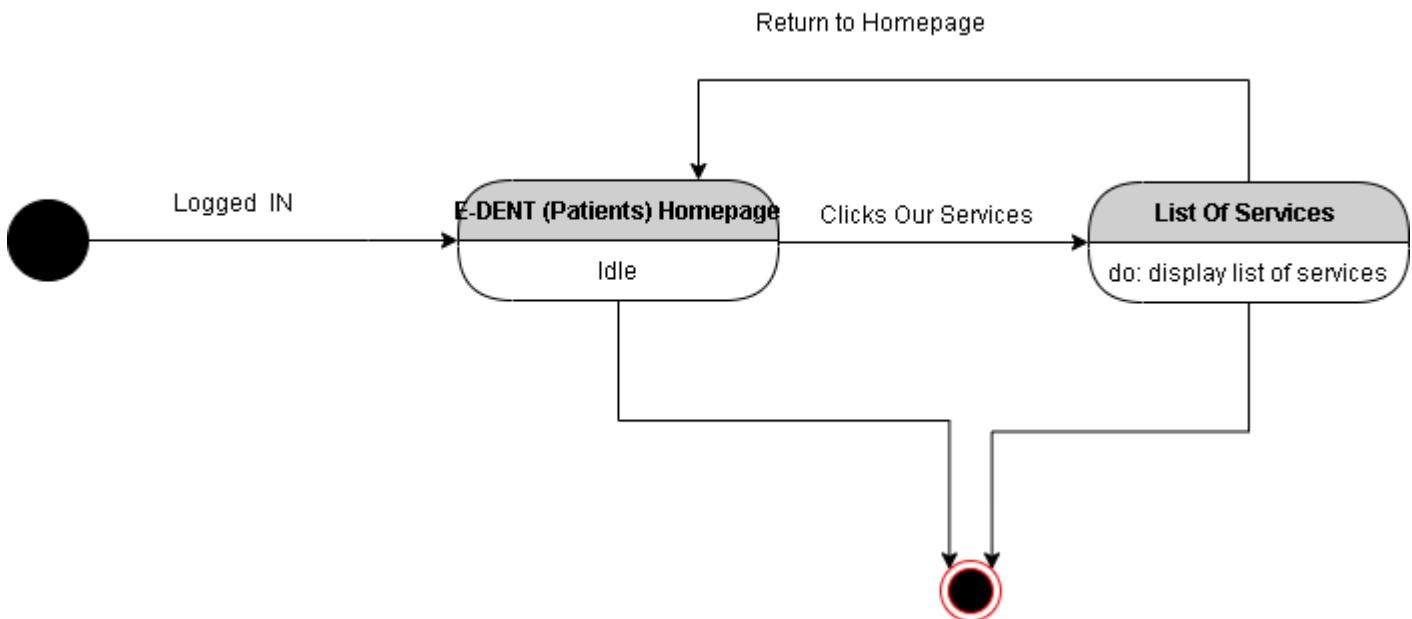


[E-DENT] Requirements Specification

ST_01.3_EditProfile

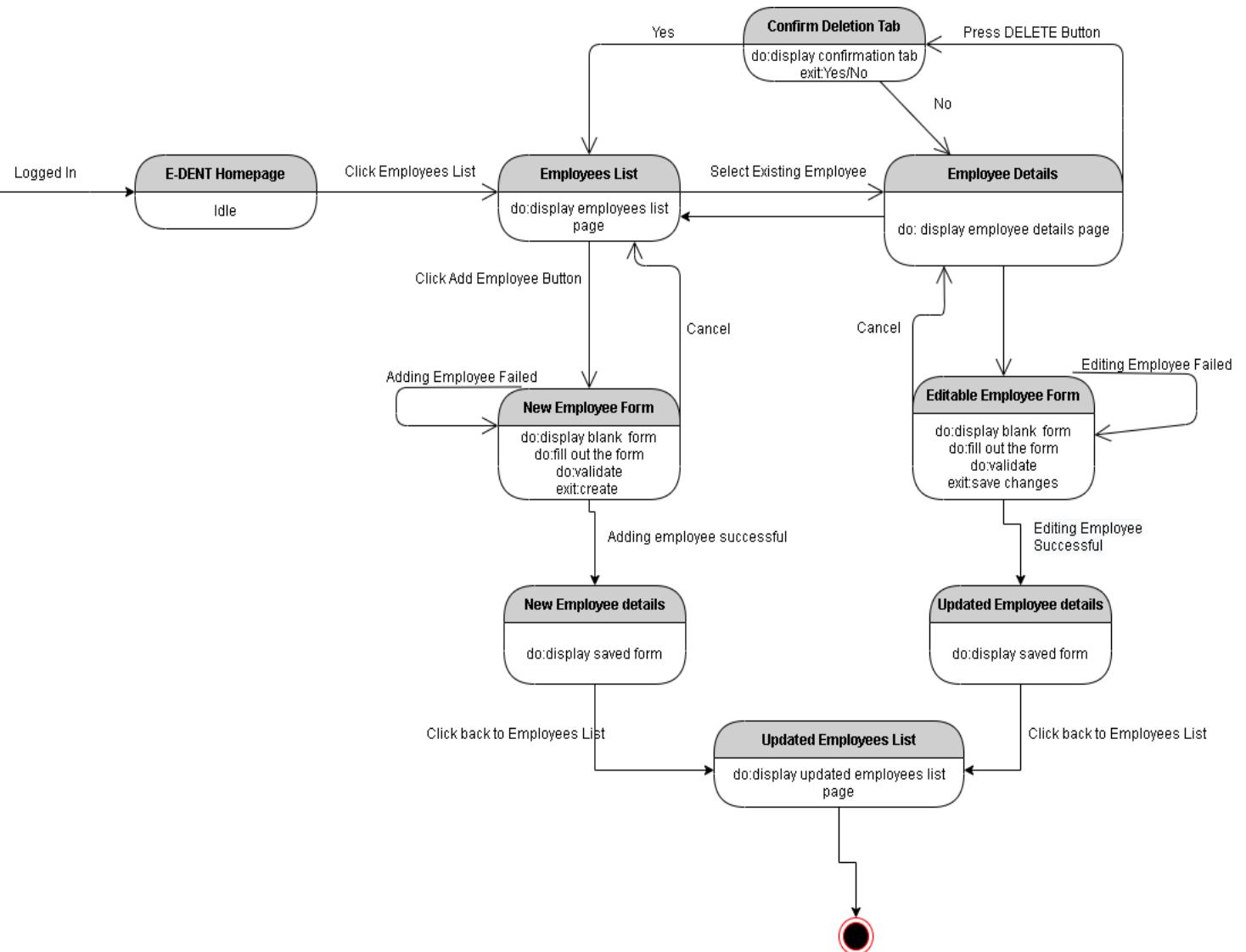


ST_04.1_ViewServices

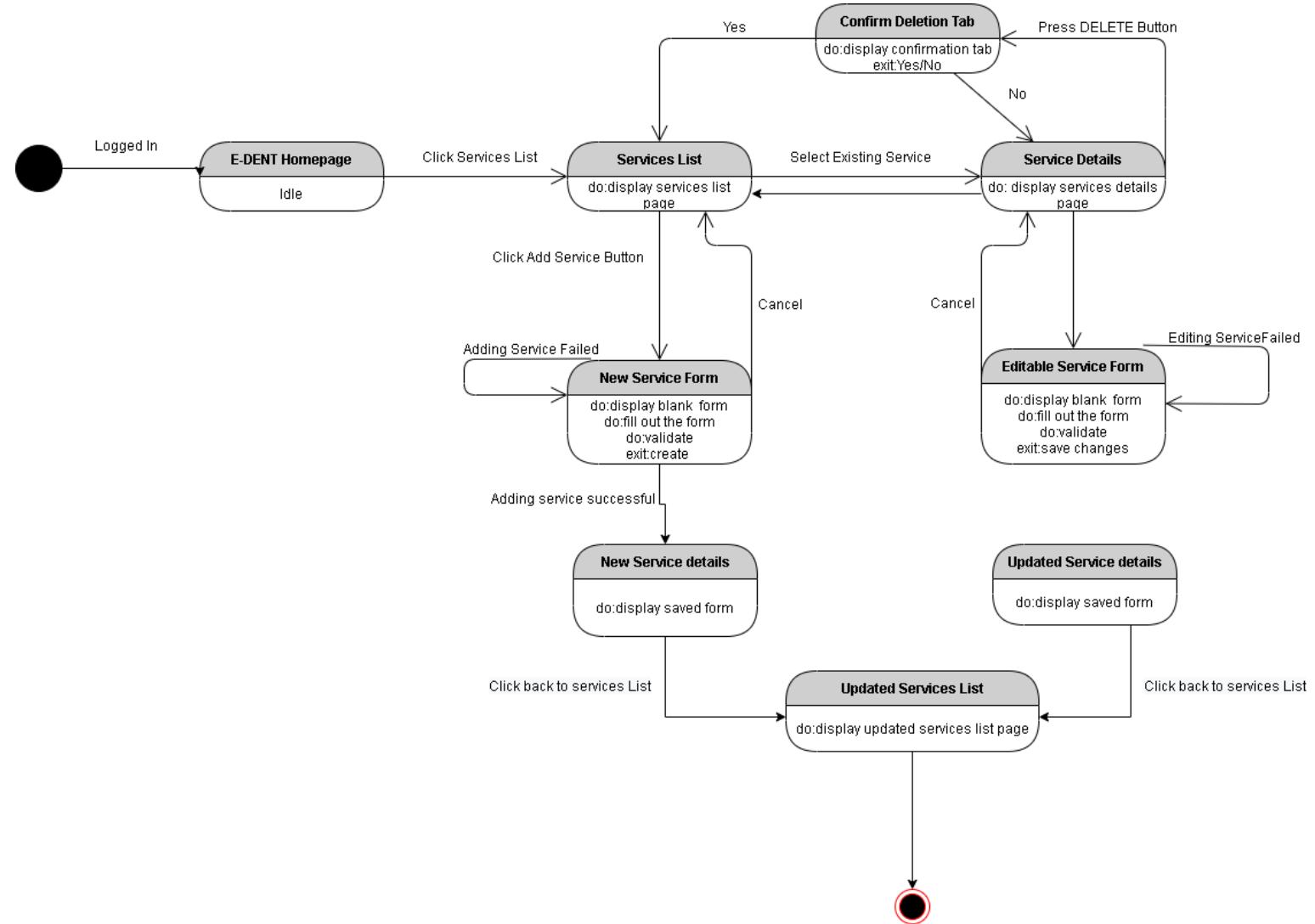


[E-DENT] Requirements Specification

ST_02.1_CRUDEmpList

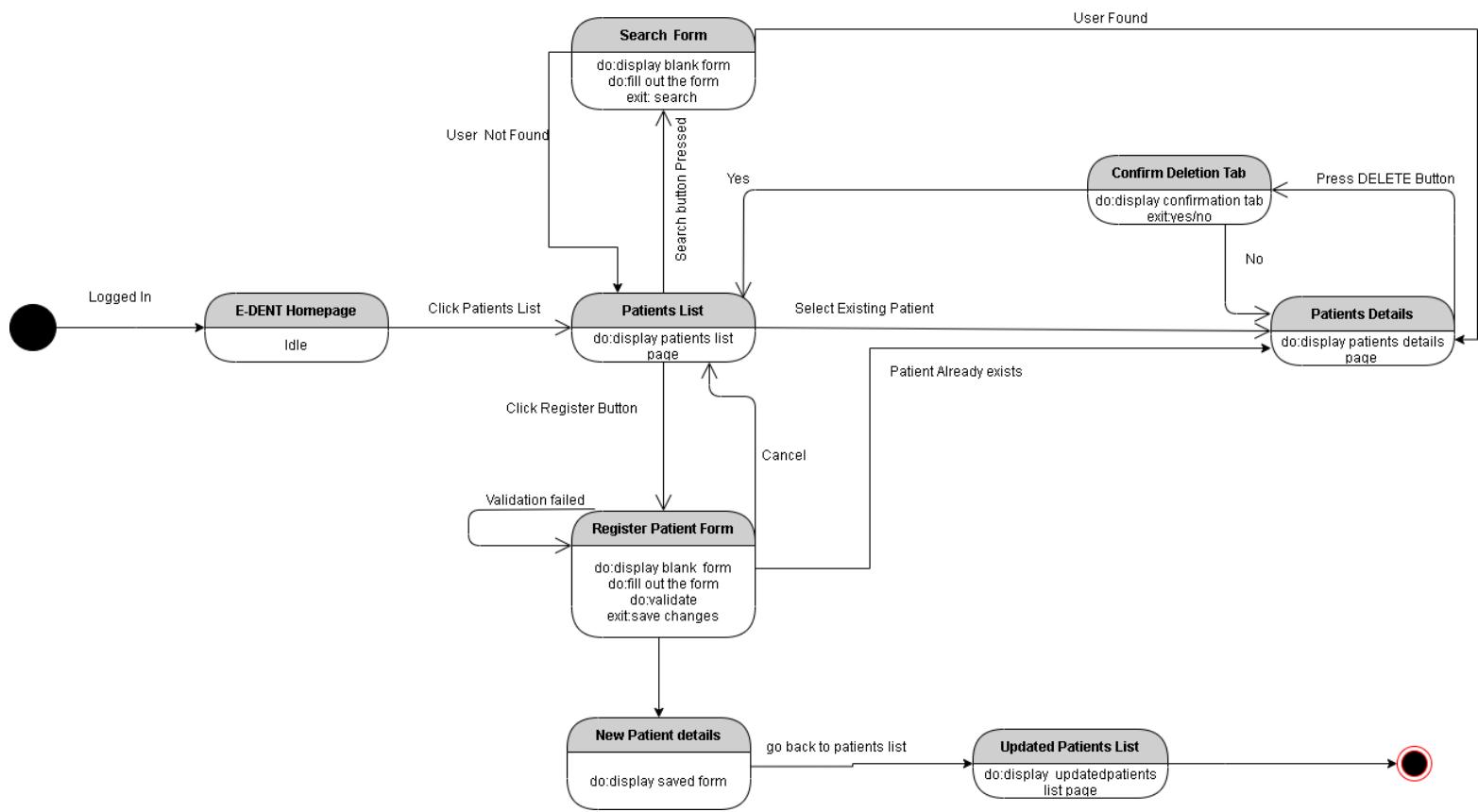


ST_02.2_CRUDServiceList

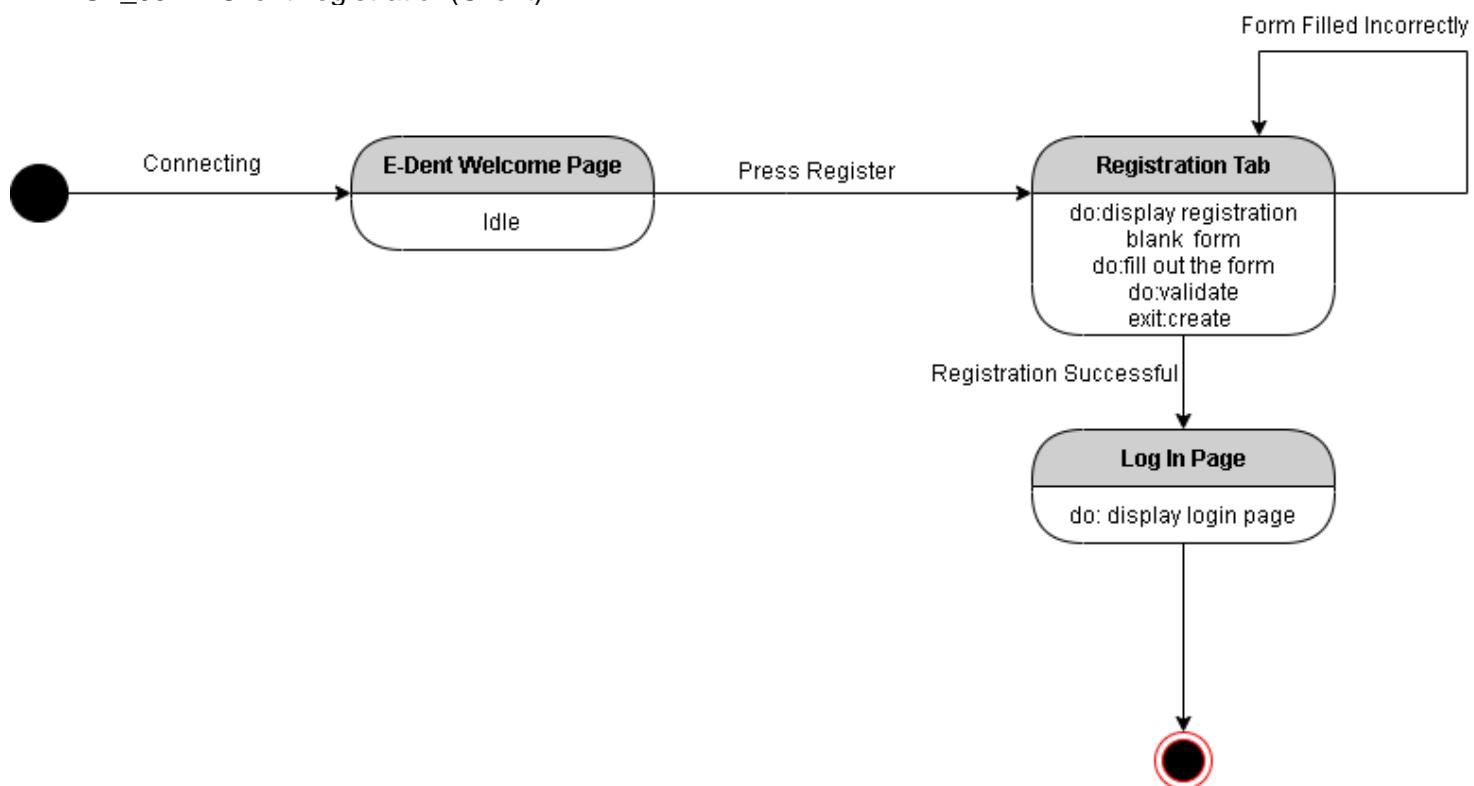


[E-DENT] Requirements Specification

ST_03.2.1-2-3PatientsAccounts(Receptionist Side)

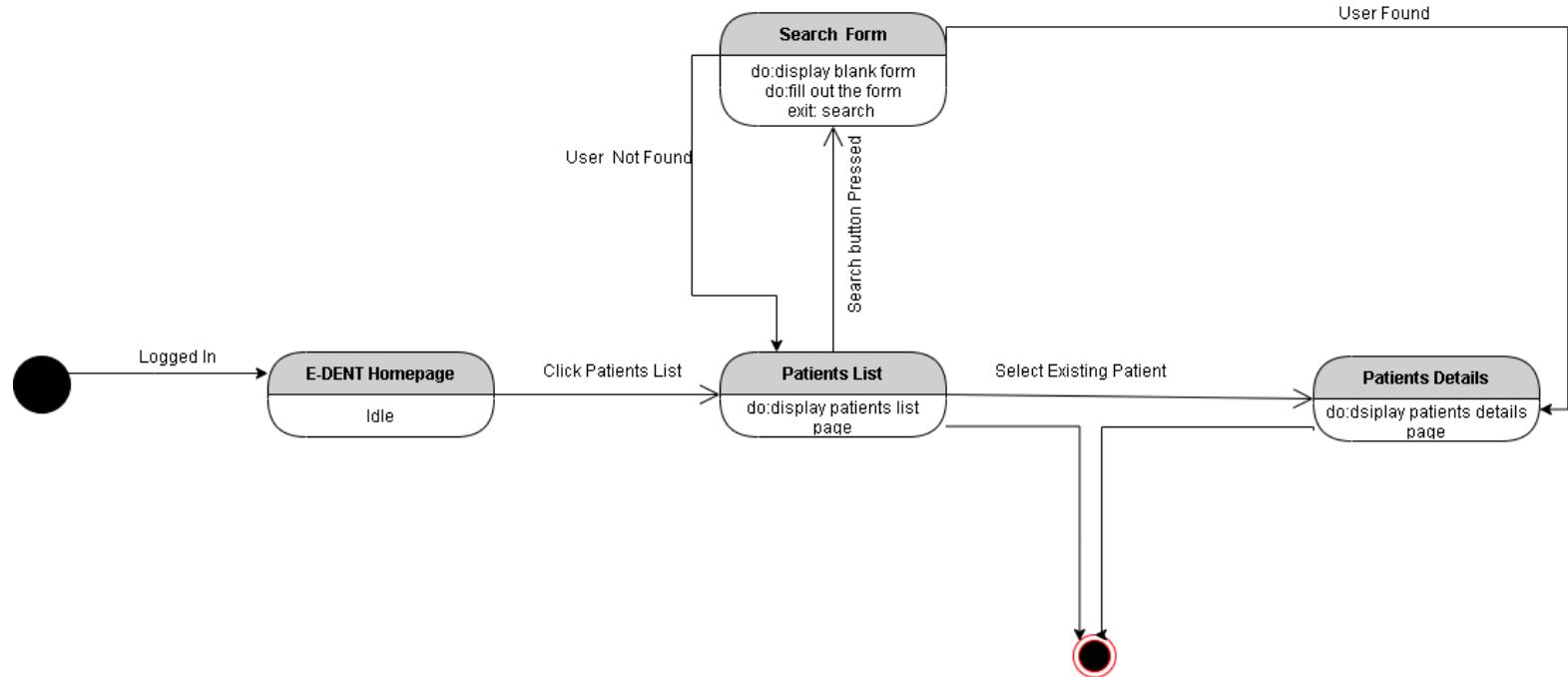


ST_03.2.1ClientRegistration(Client)

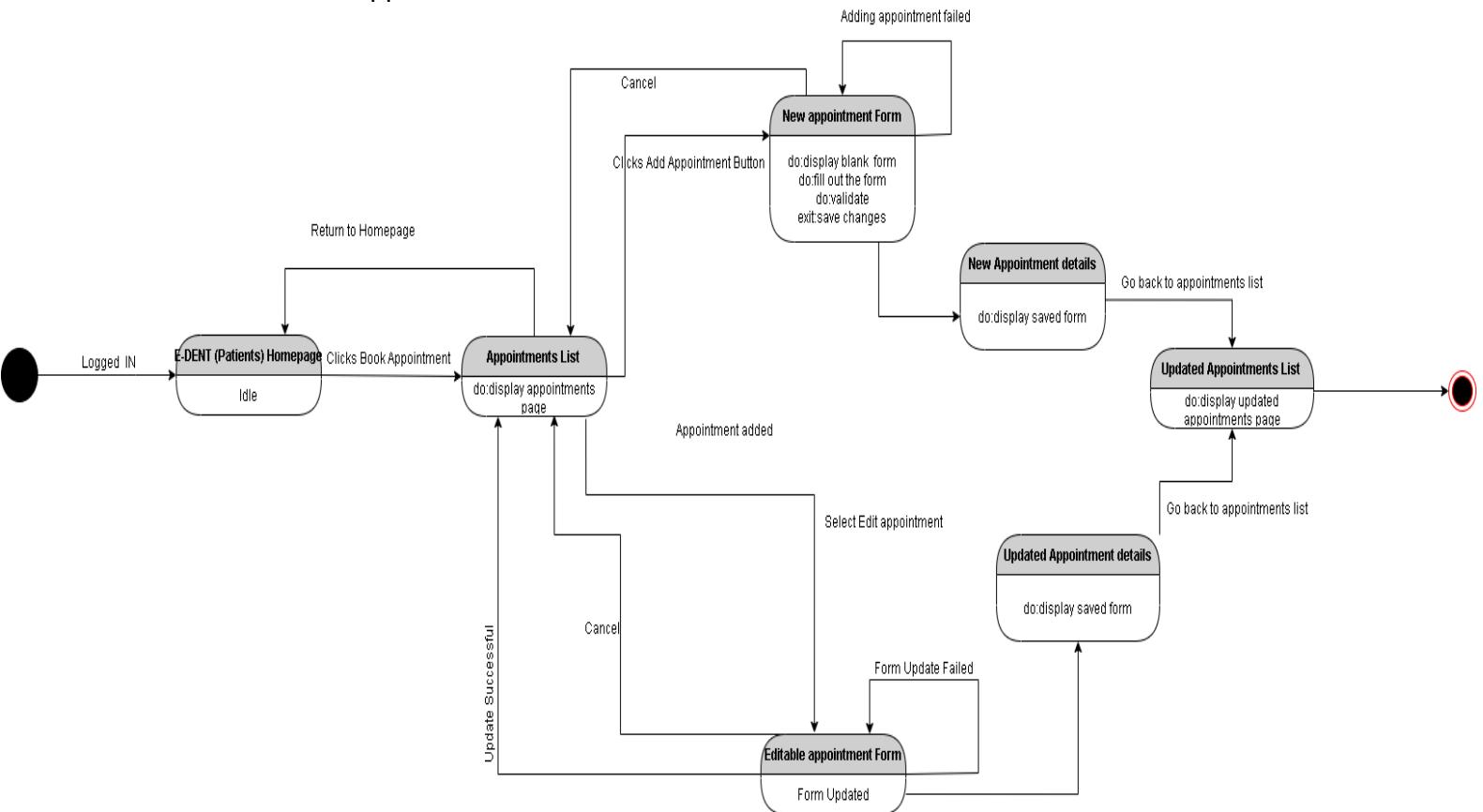


[E-DENT] Requirements Specification

ST_03.2.3ClientSearch

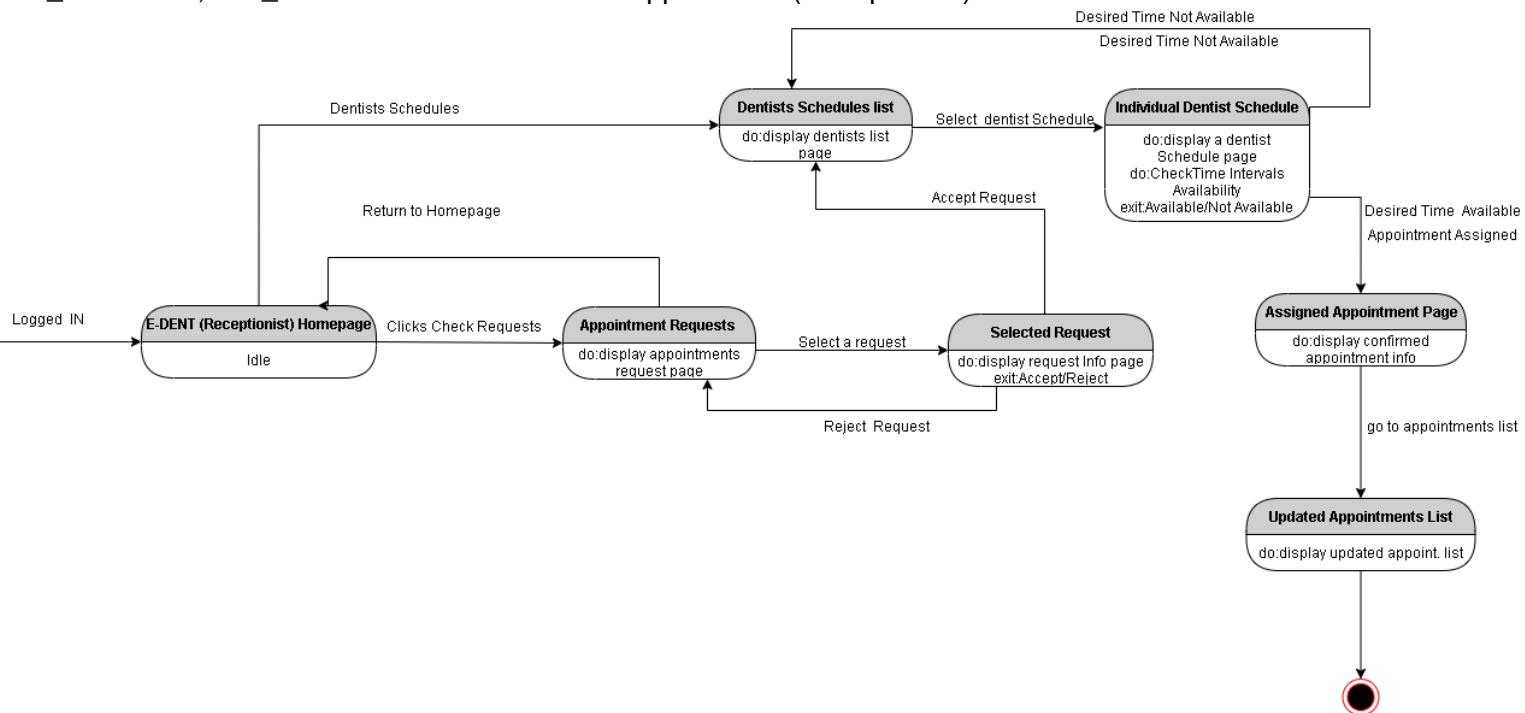


ST_04.2.1-2_Add&CheckAppointment

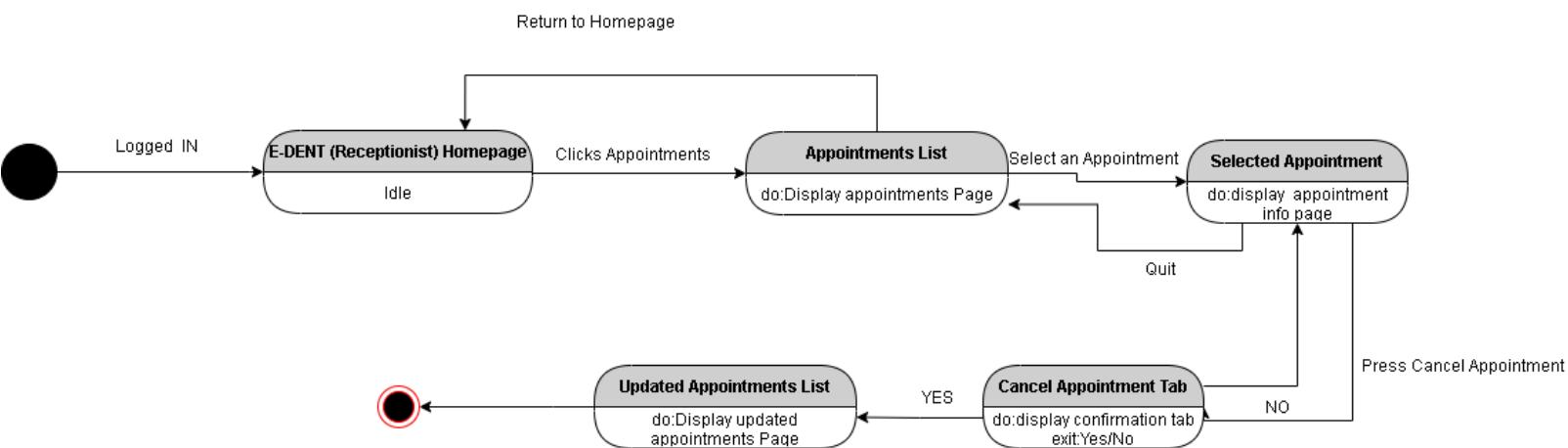


[E-DENT] Requirements Specification

ST_05.1.1-2-3;06.1_DentistSched&Add&CheckAppointment(Receptionist)

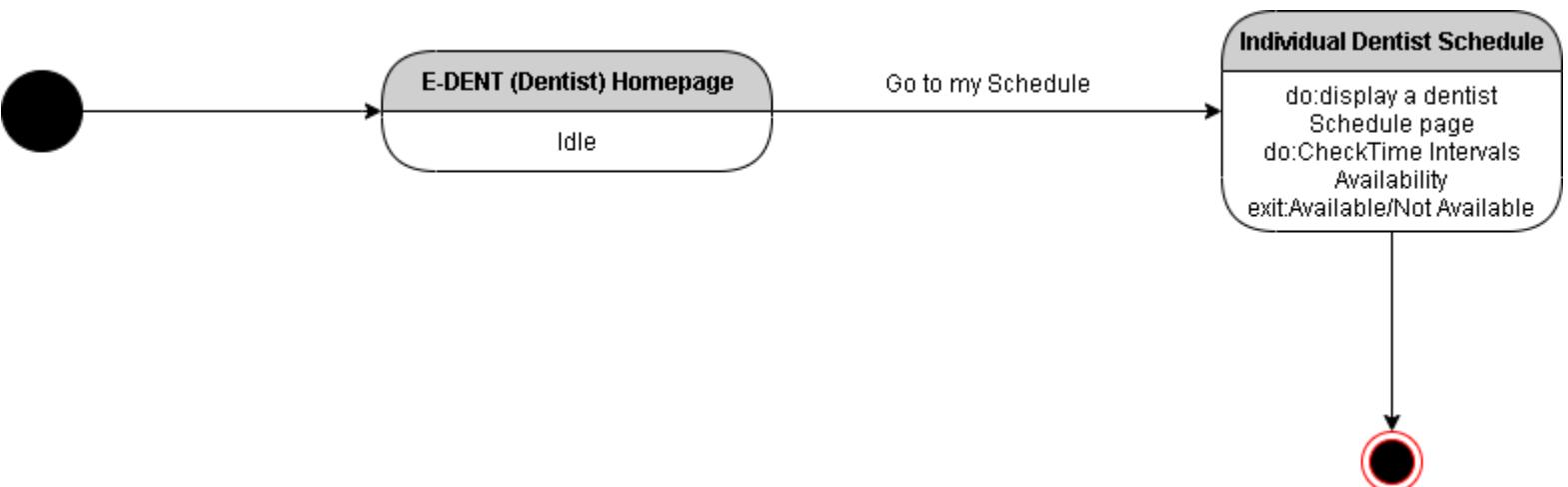


ST_05.1.3_CancelAppointment(Receptionist)

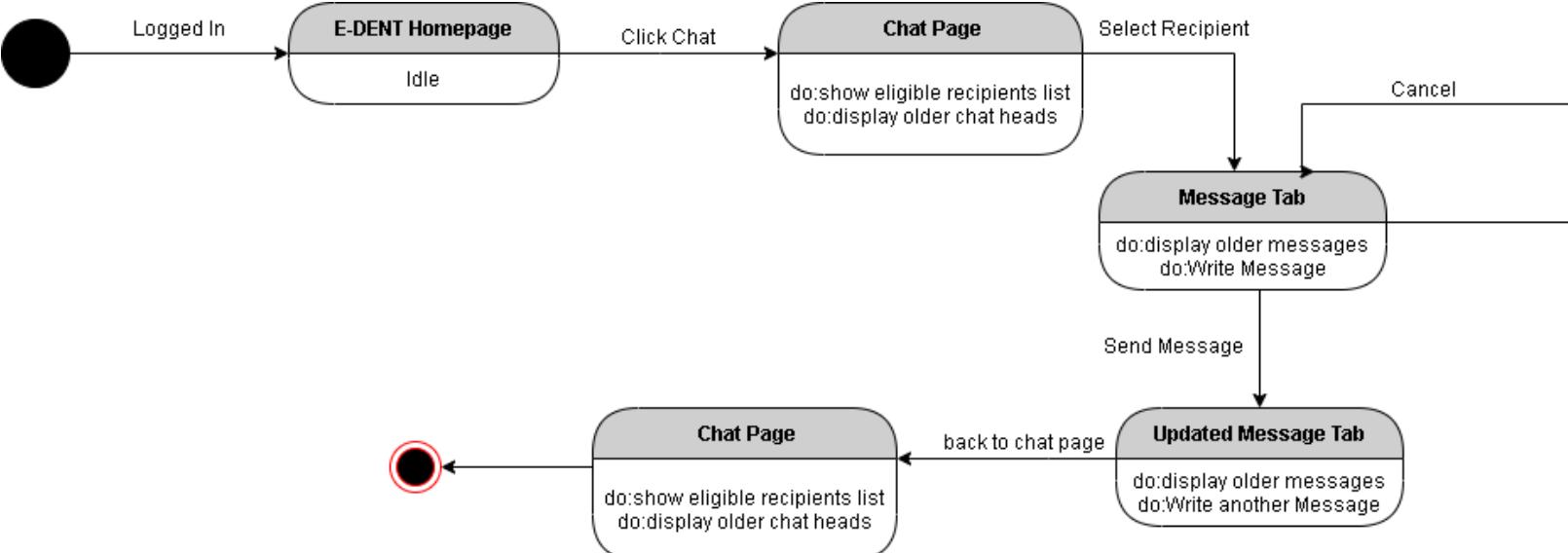


[E-DENT] Requirements Specification

ST_06.1_ViewDentistSchedule(Dentist)

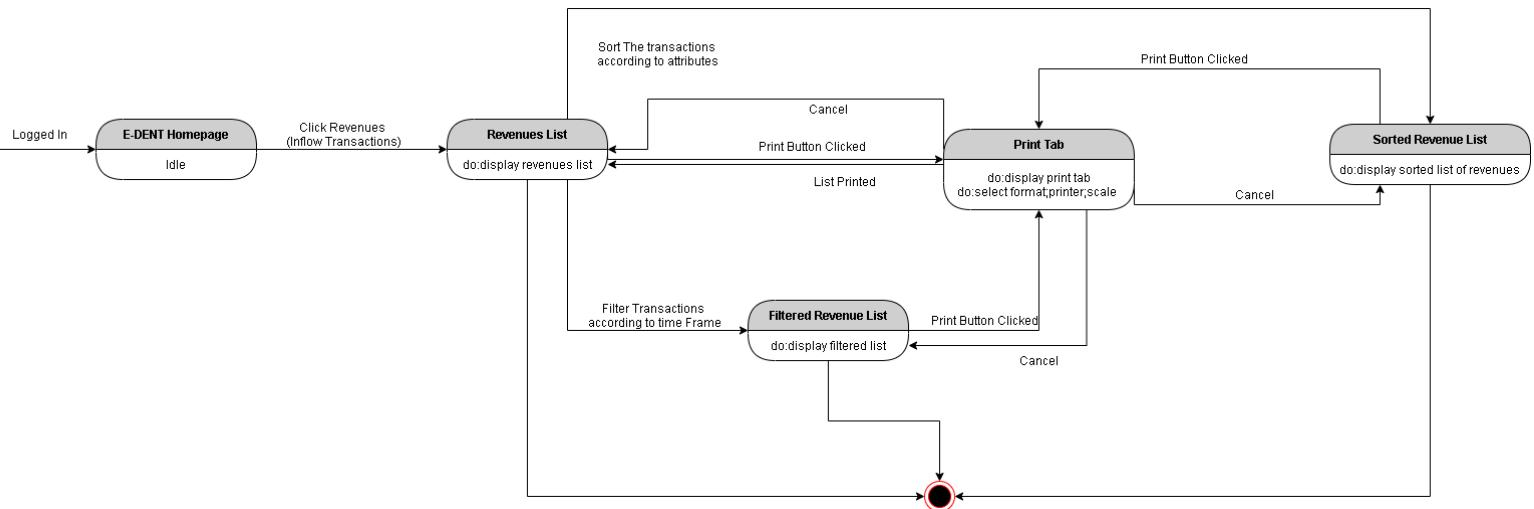


ST_06.3_Chat

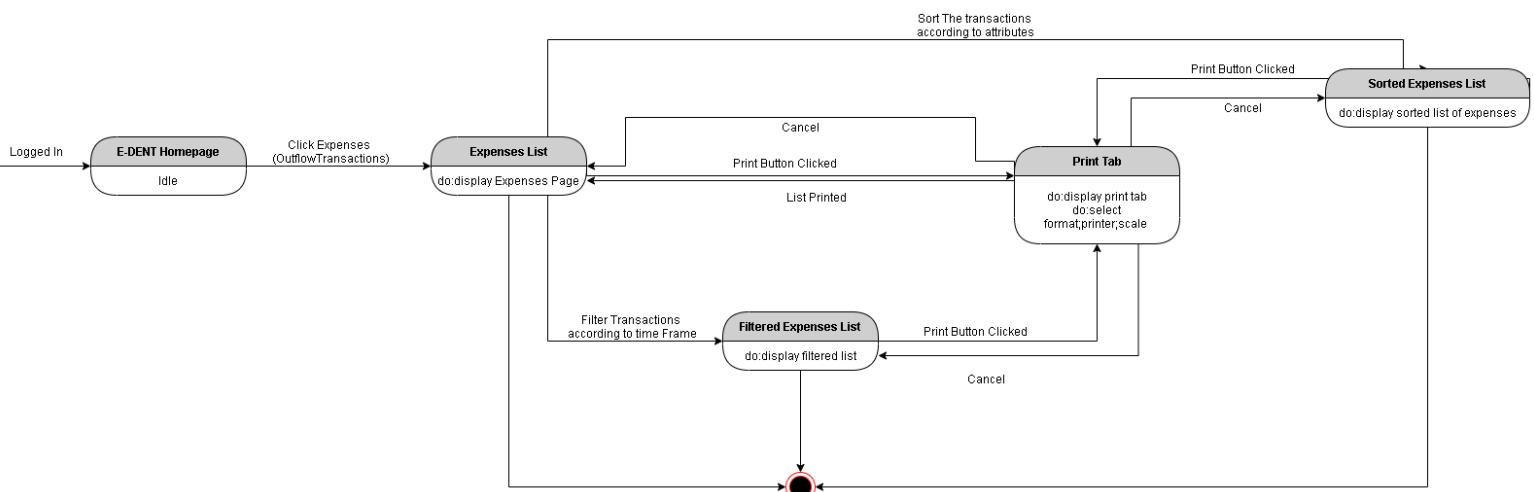


[E-DENT] Requirements Specification

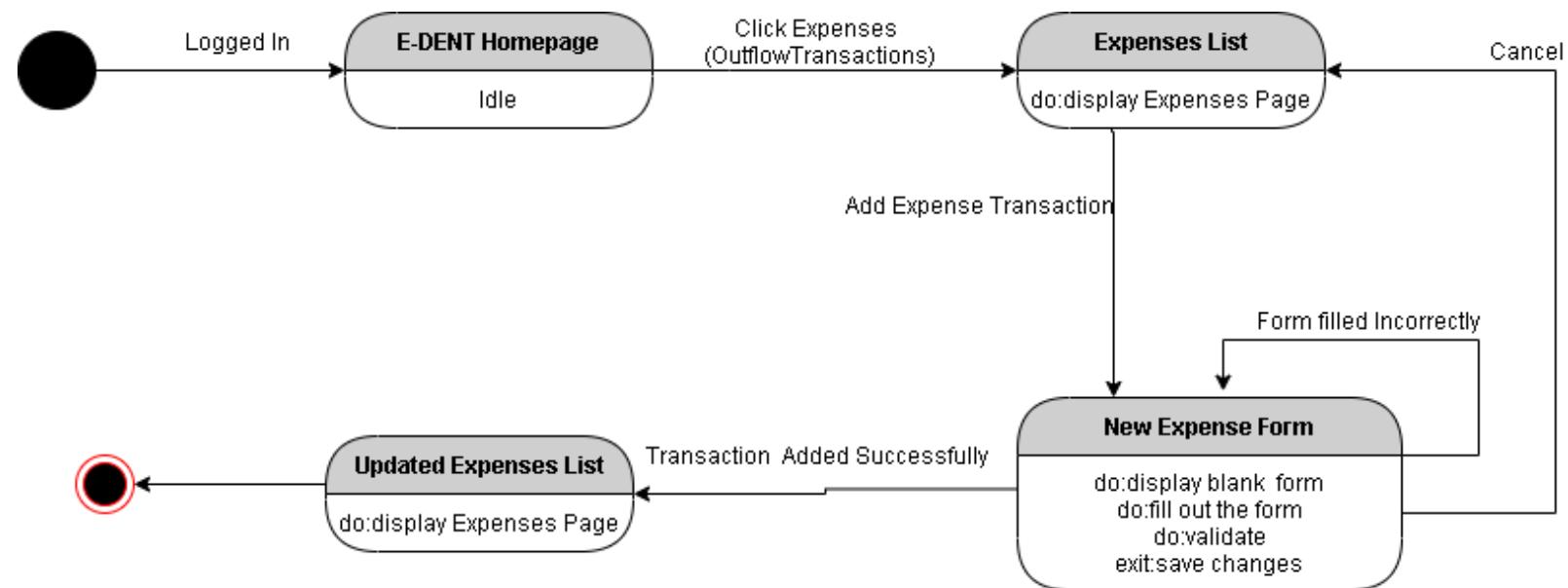
ST_07.1.1-2-3-4_InflowTrans-View-Filter-Sort-Print



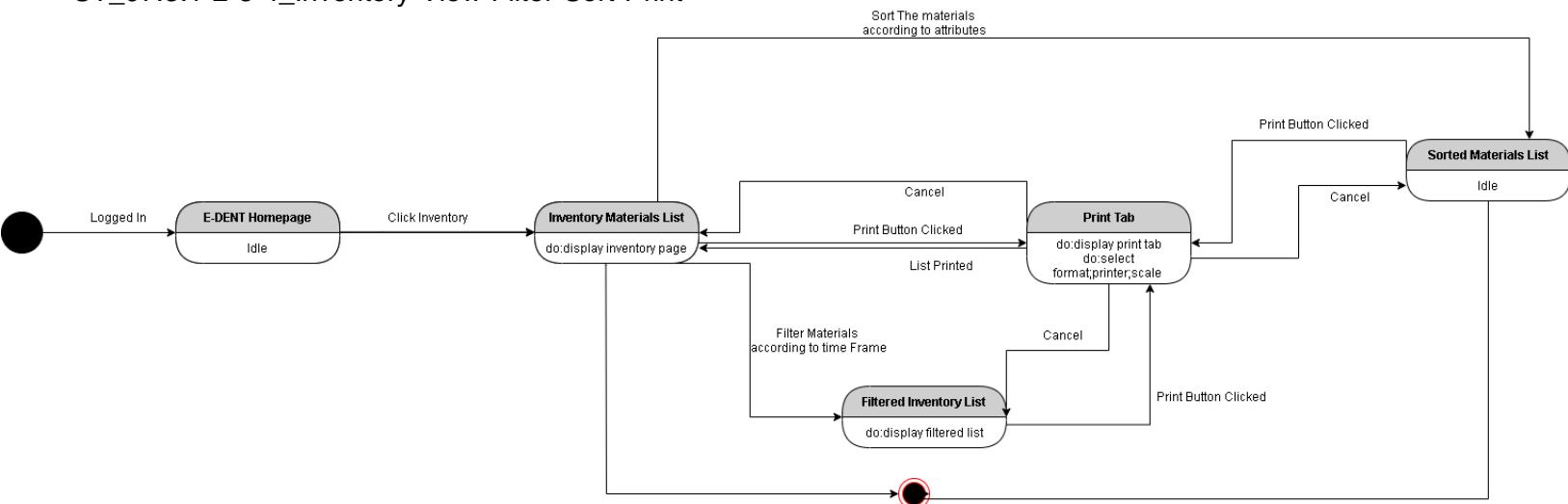
ST_07.2.1-2-3-4_ExpTrans-View-Filter-Sort-Print



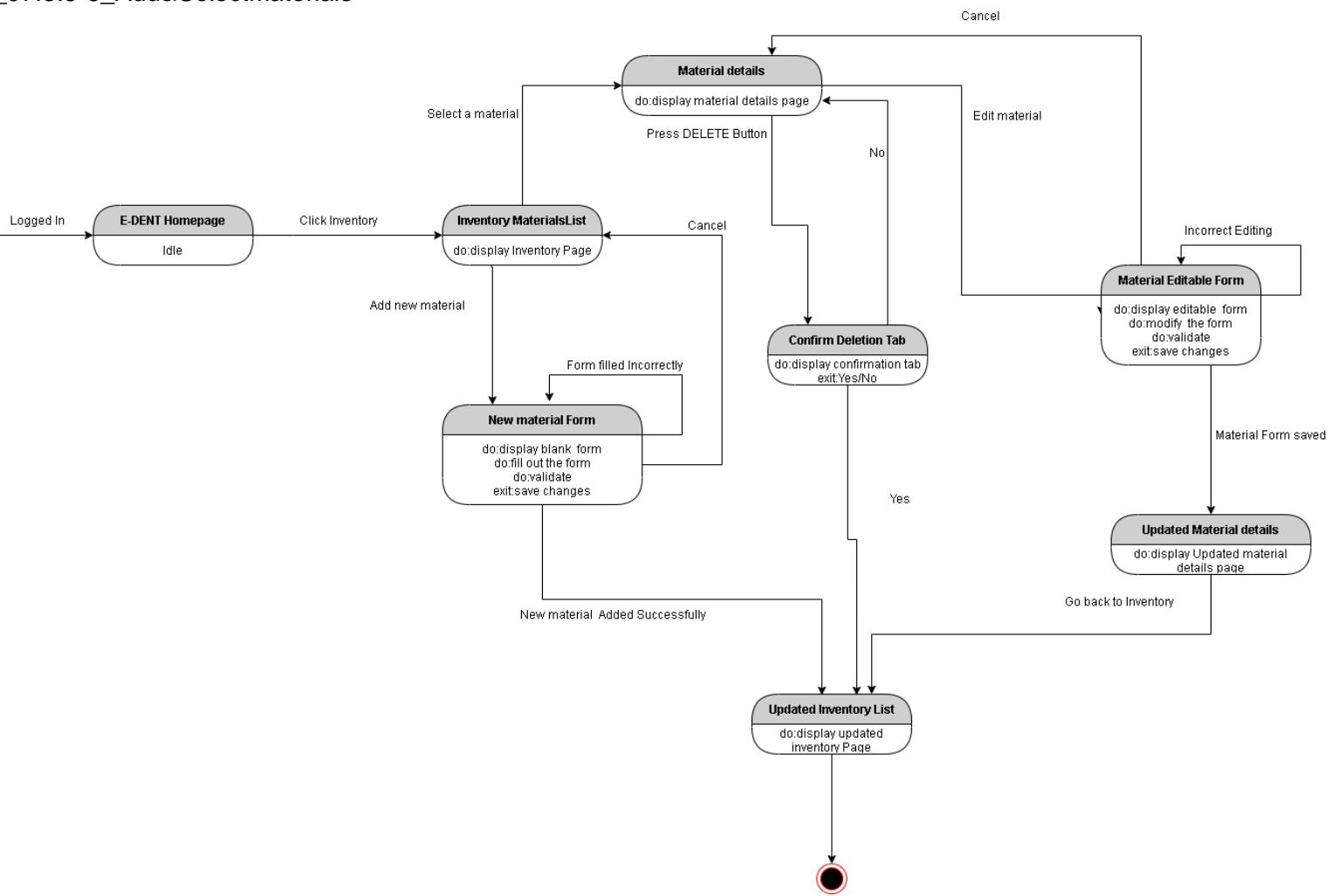
ST_07.2.5_AddExpTransaction



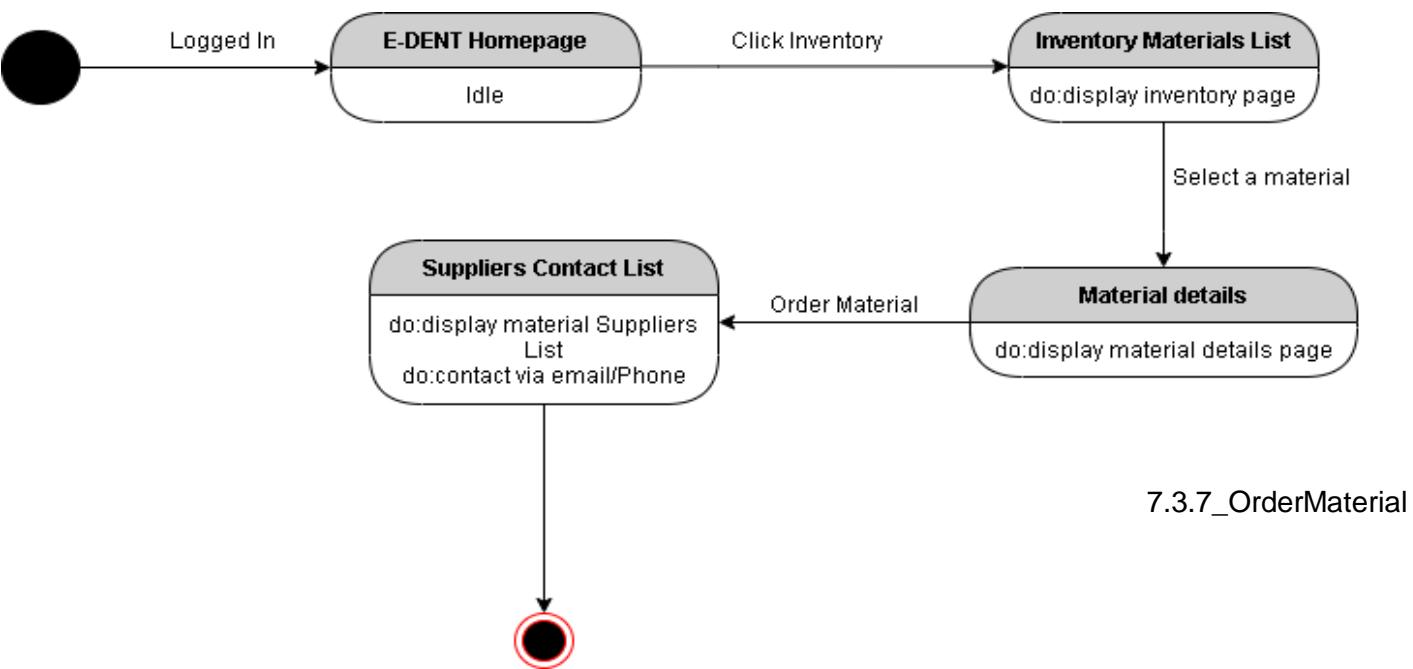
ST_07.3.1-2-3-4_Inventory-View-Filter-Sort-Print



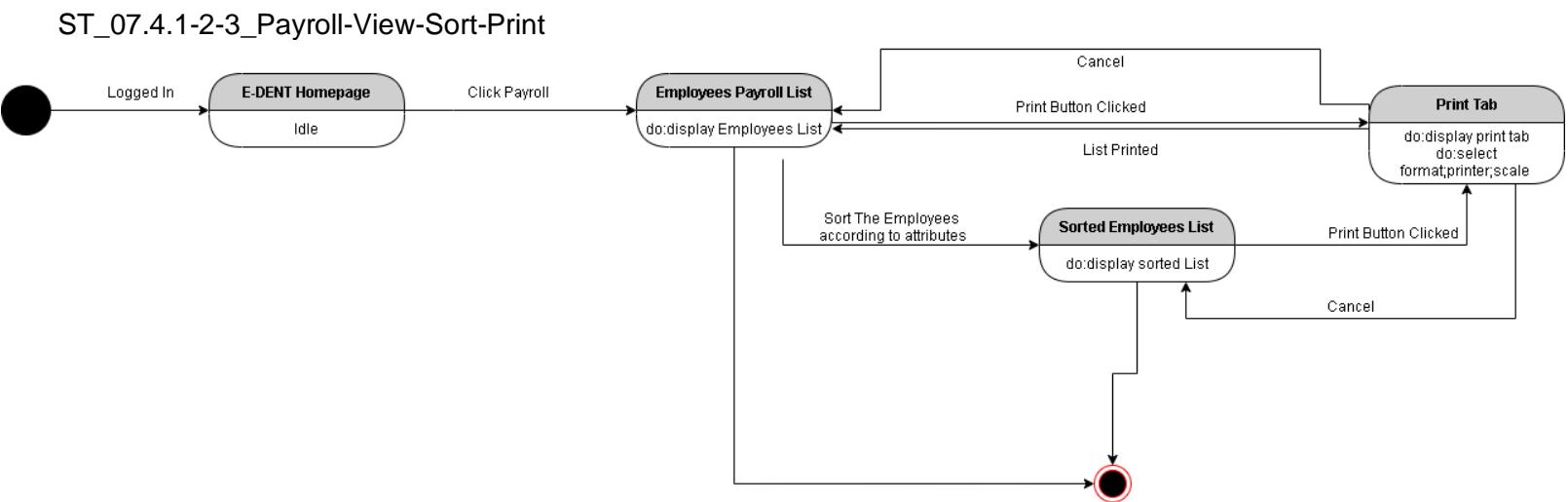
ST_07.3.5-6_Add&SelectMaterials



[E-DENT] Requirements Specification

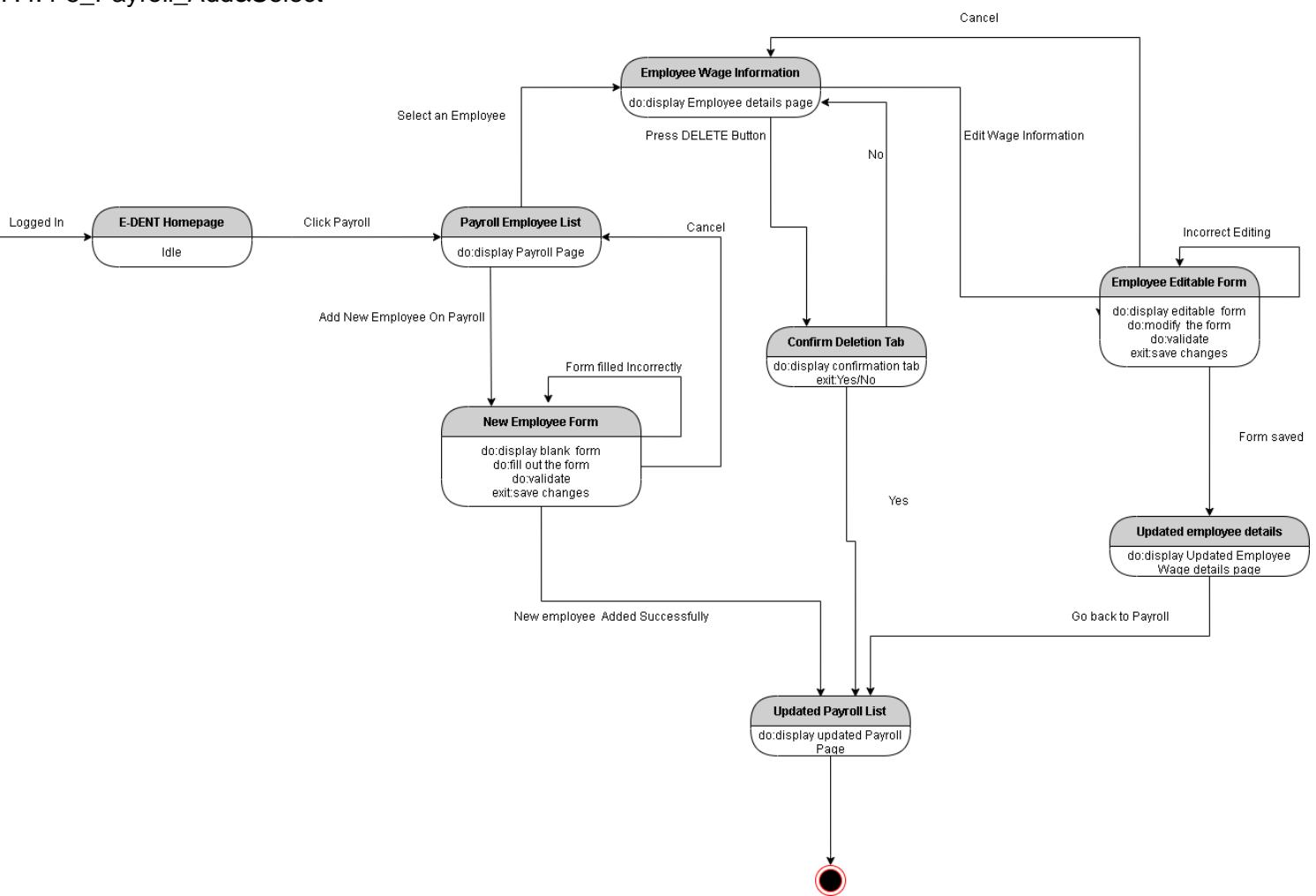


7.3.7_OrderMaterial

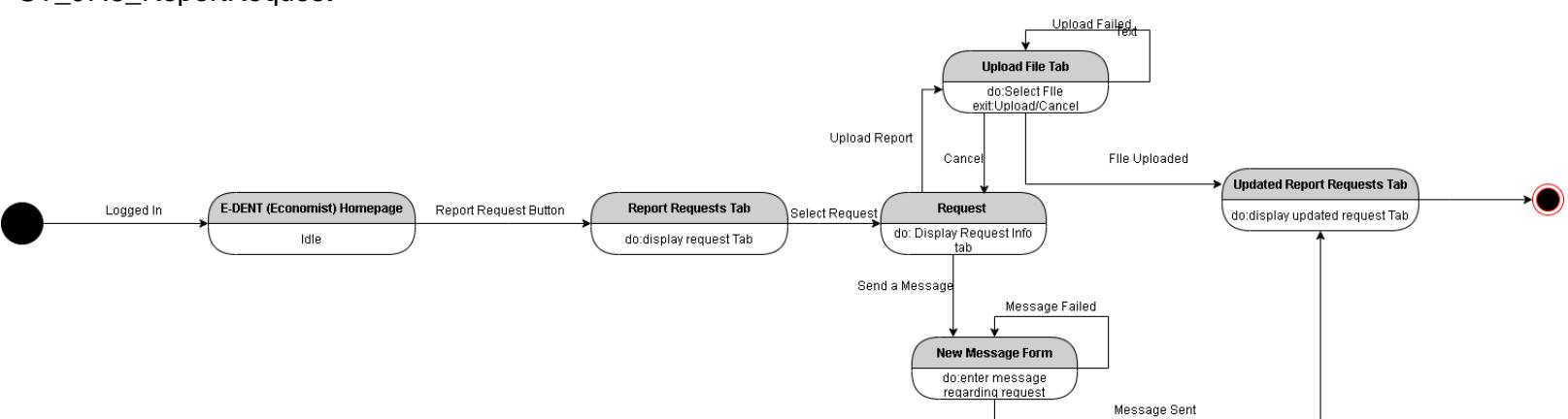


[E-DENT] Requirements Specification

ST_07.4.4-5_Payroll_Add&Select

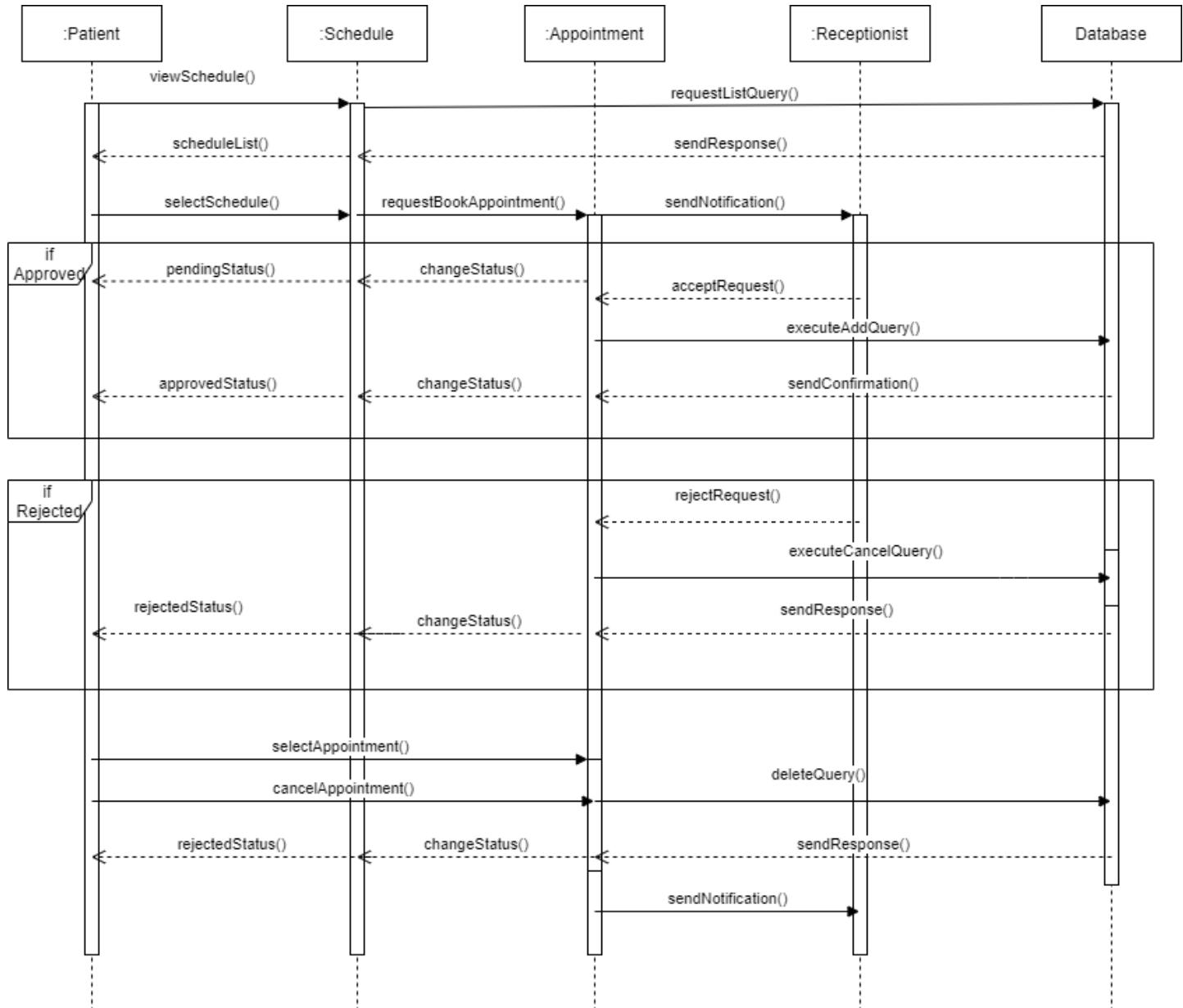


ST_07.5_ReportRequest

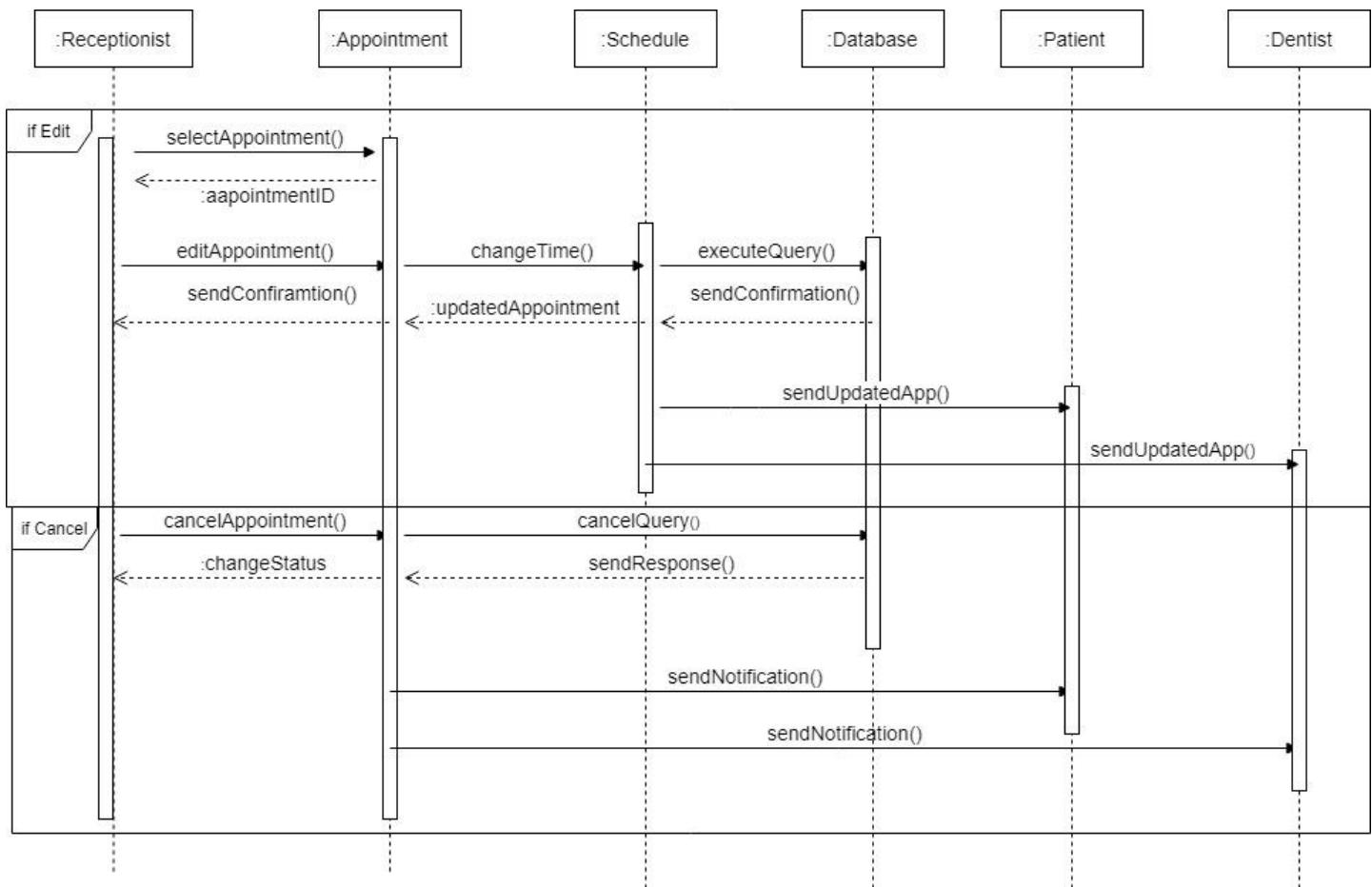


Sequence Diagrams

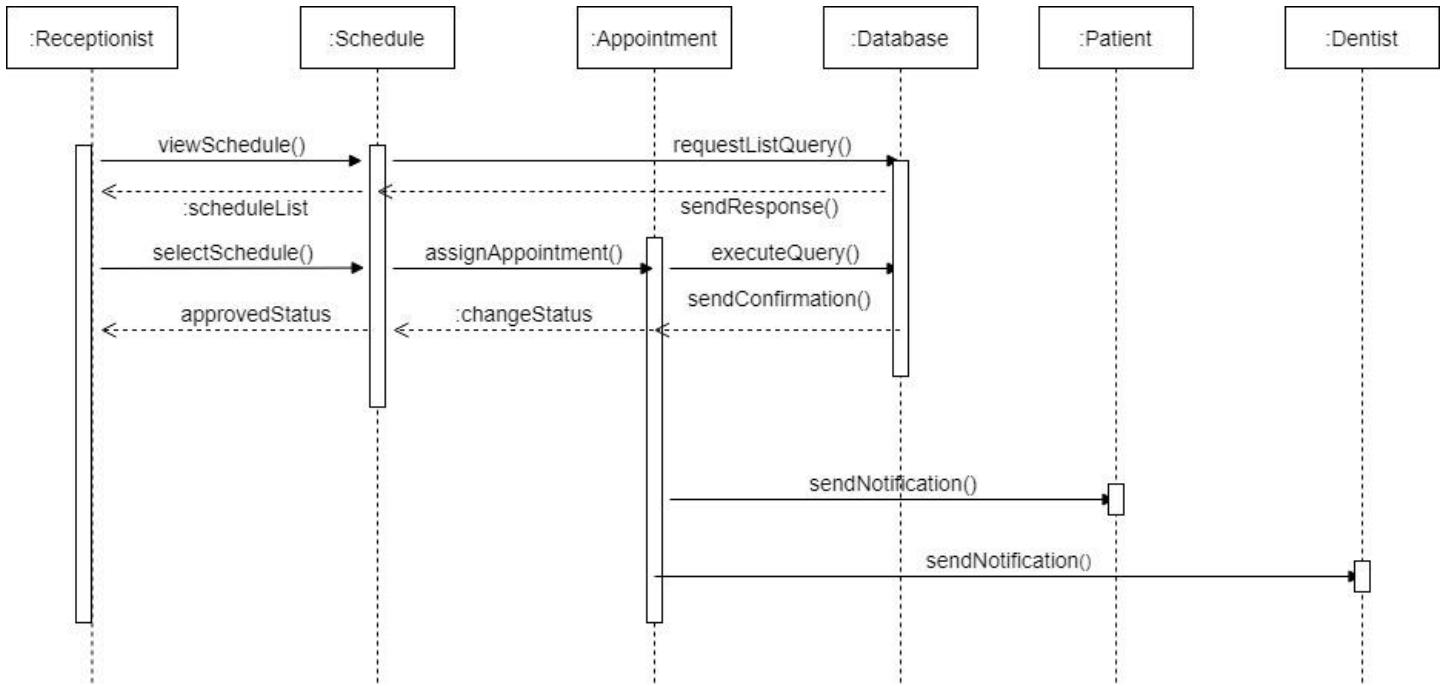
Seq_4.2.1-4.2.2-5.1.2_Book Appointment



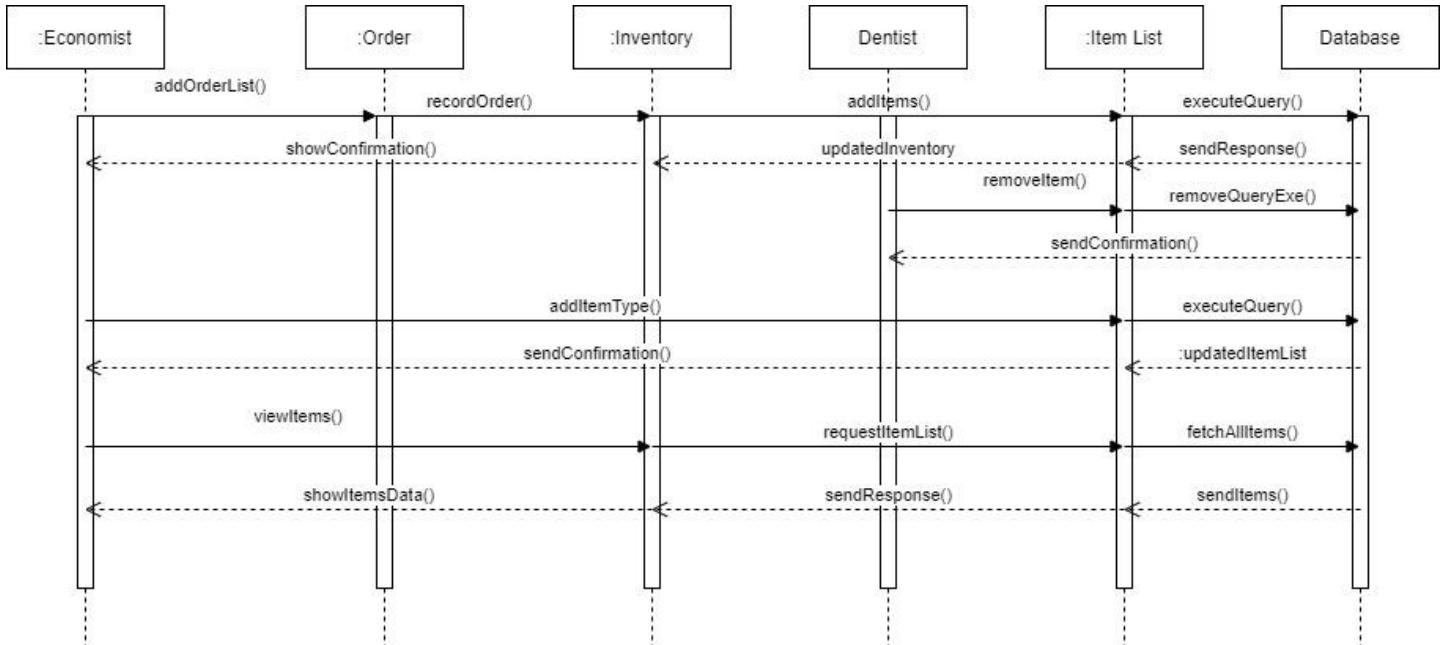
Seq_5.1.1-5.1.3-5.2_Edit Appointment



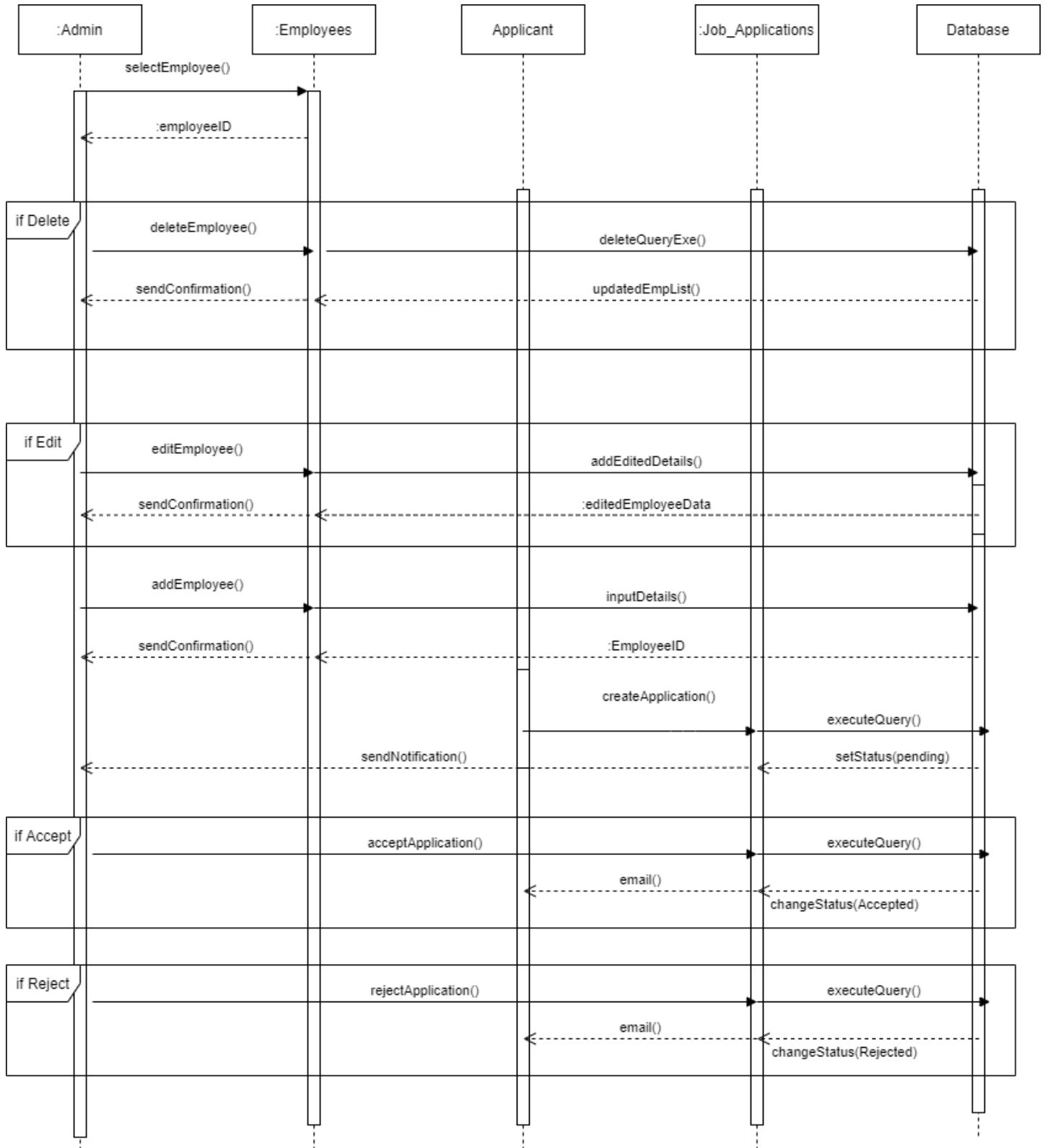
Seq_5.2-5.1.2_AssignAppointment



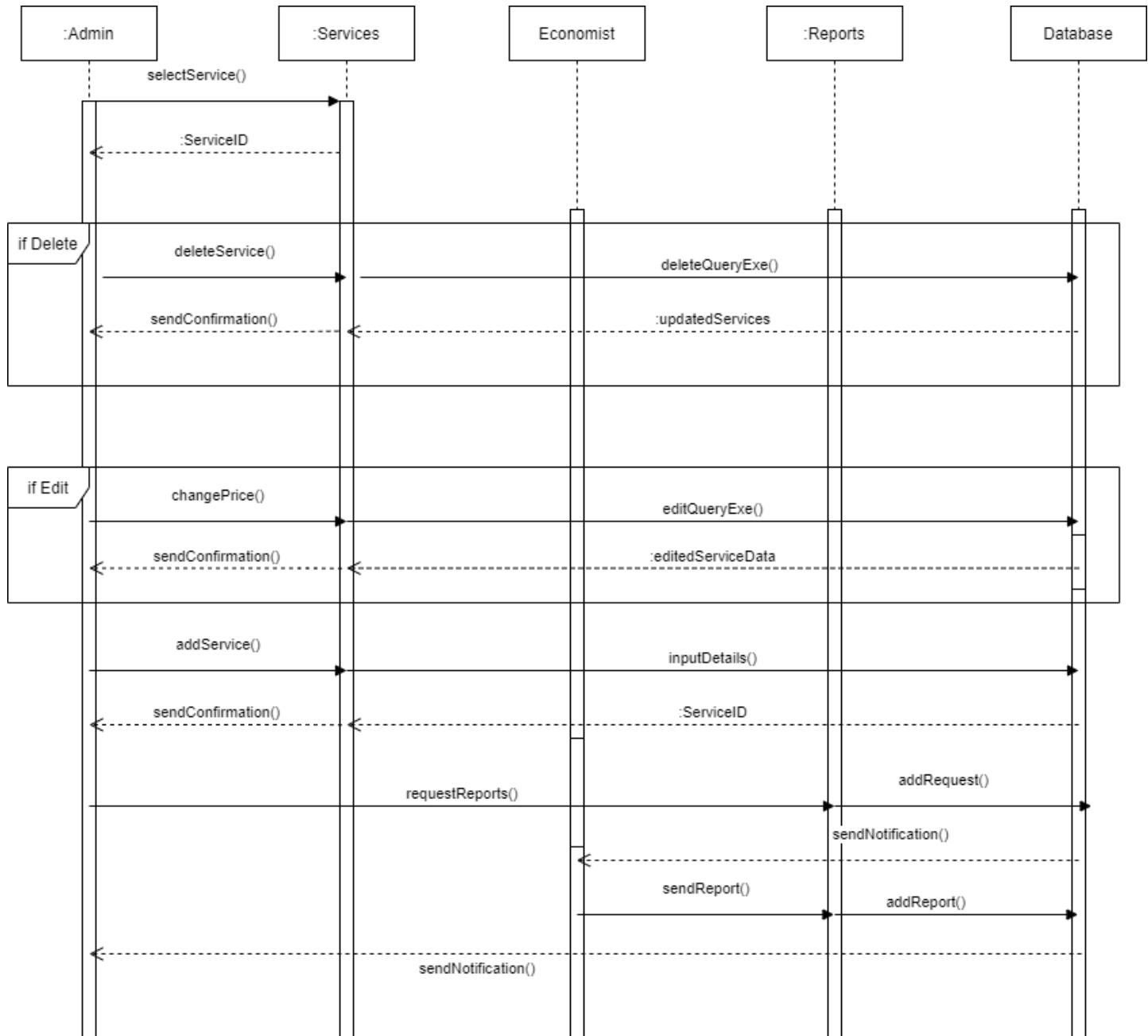
Seq_7.3.7-6.4-7.3.1_ManageInventory



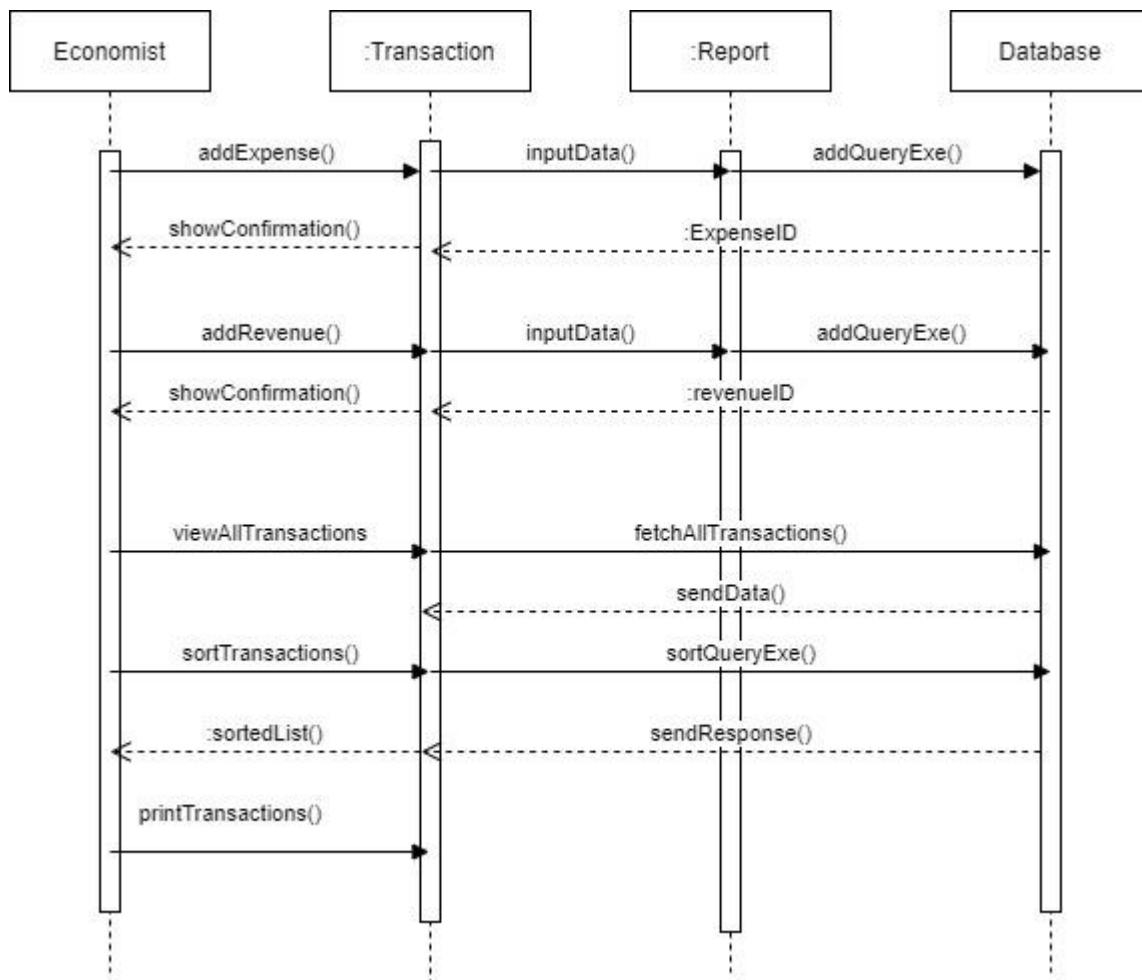
Seq_2.4-2.1-3.1Emp-JobApp



Seq_2.3-2.2_Services_FR

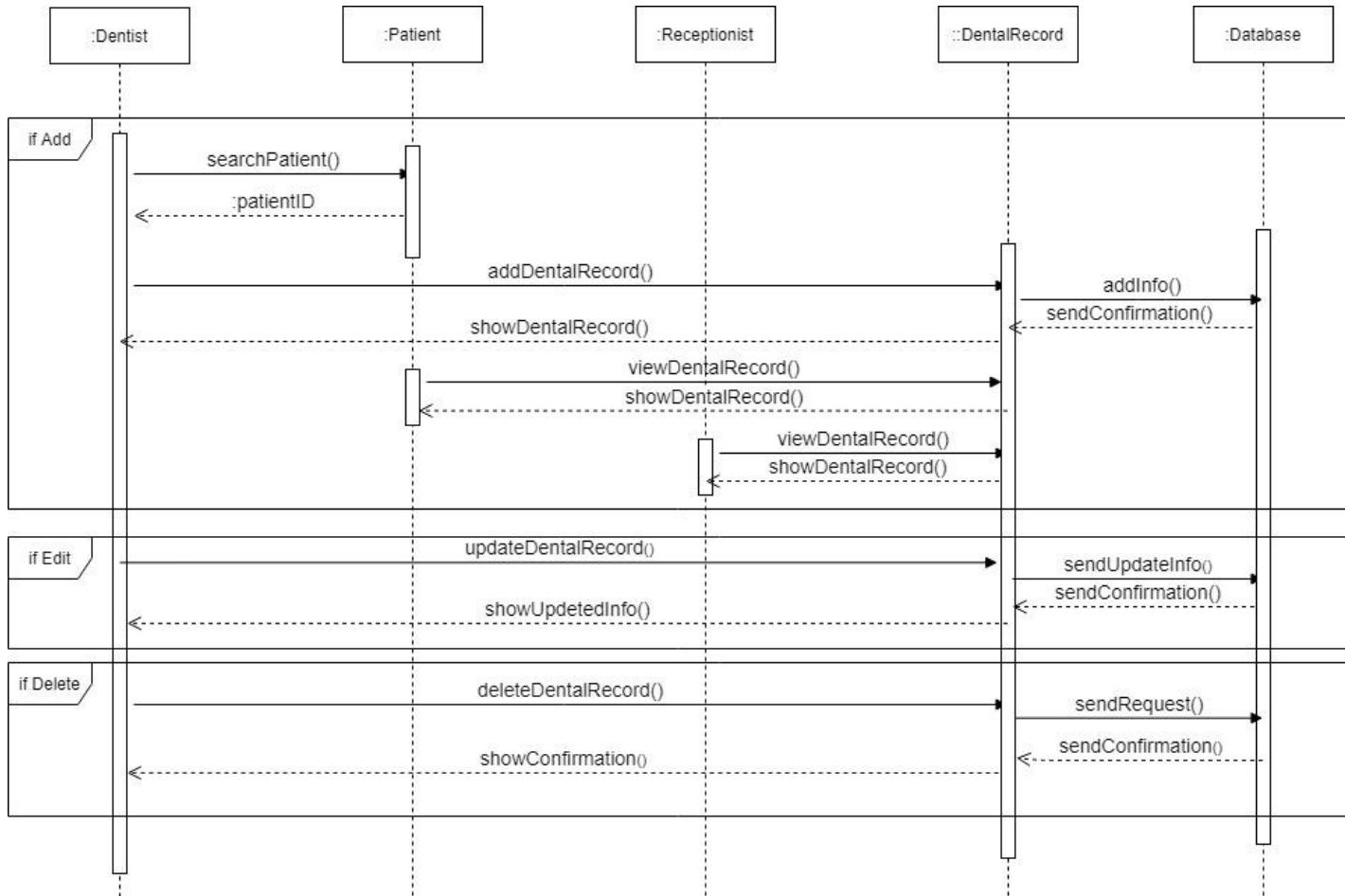


Seq_7.2.4-7.1.1-7.1.4ManageTransactions



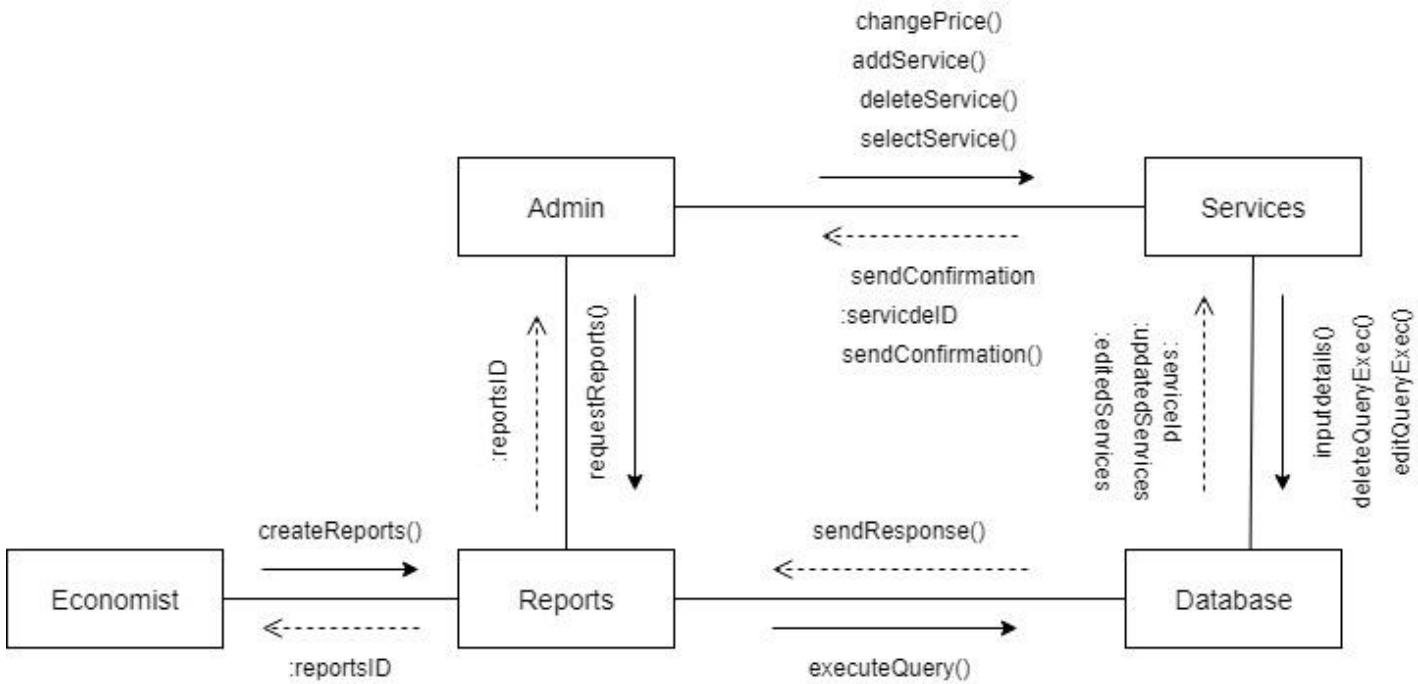
[E-DENT] Requirements Specification

Seq_3.2.3-6.2-6.3ManageDR

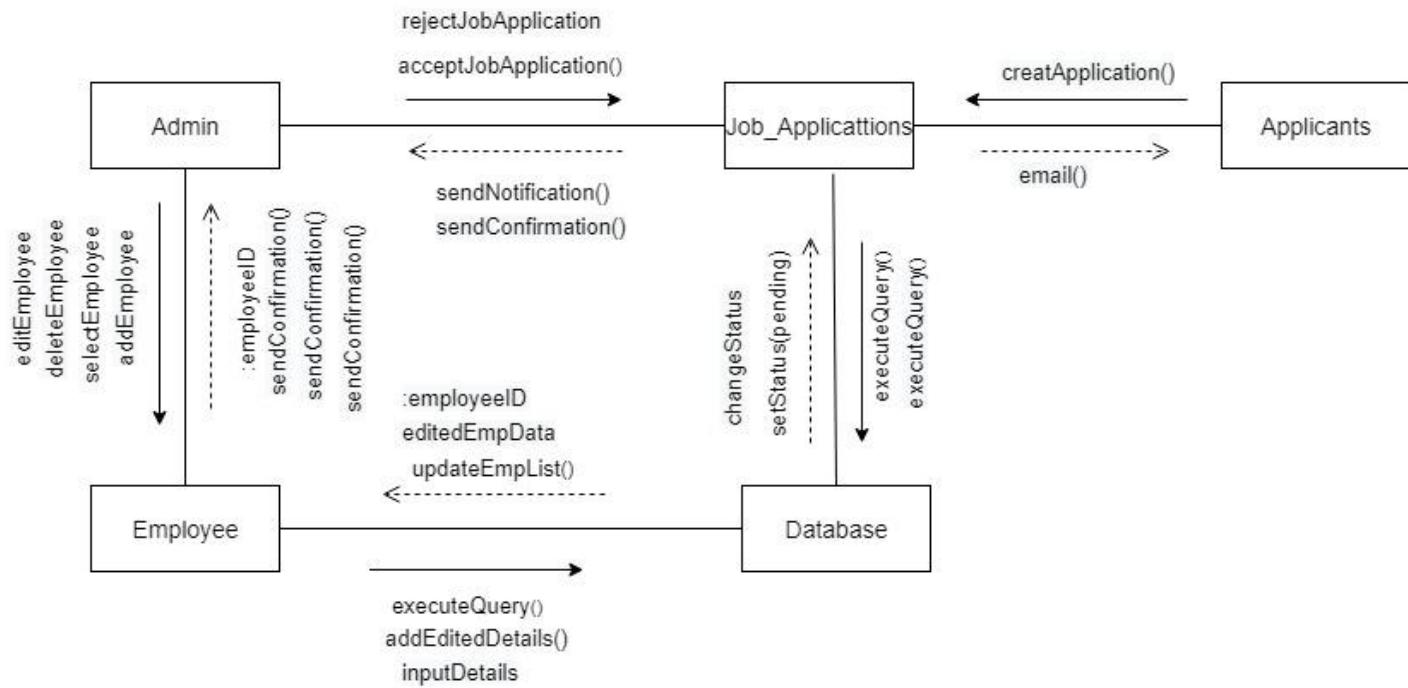


[E-DENT] Requirements Specification
COLLABORATION DIAGRAMS

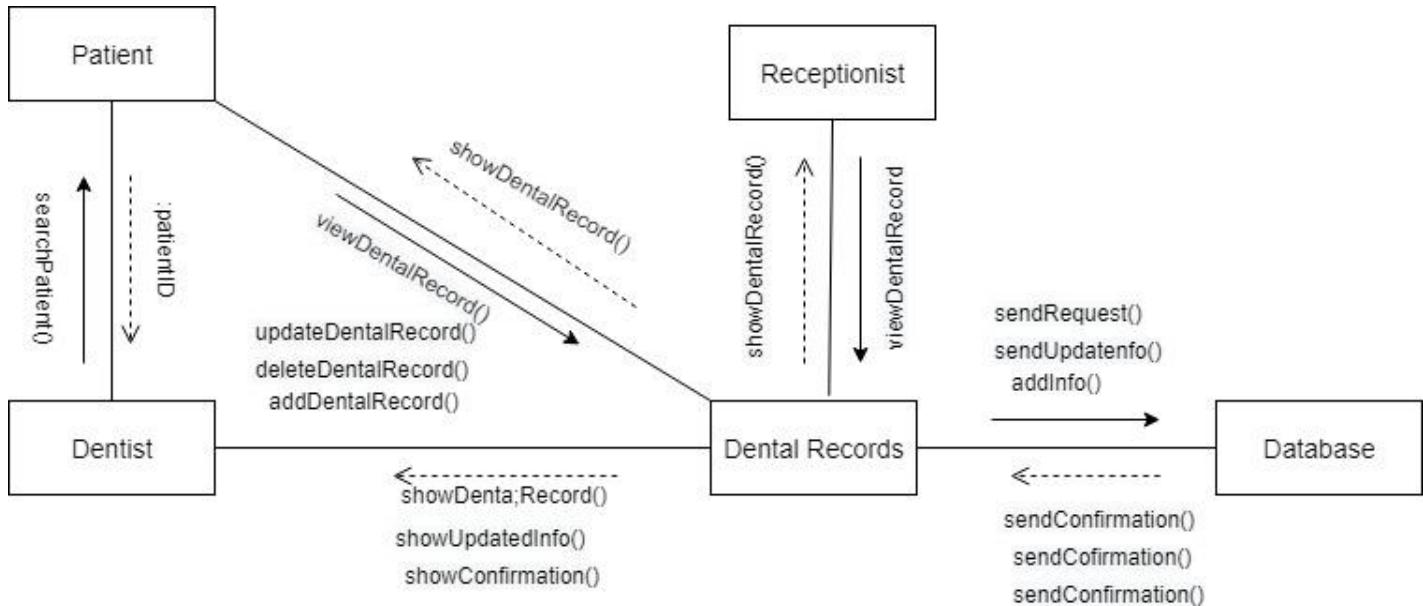
Col_2.3-2.2_Service_FR



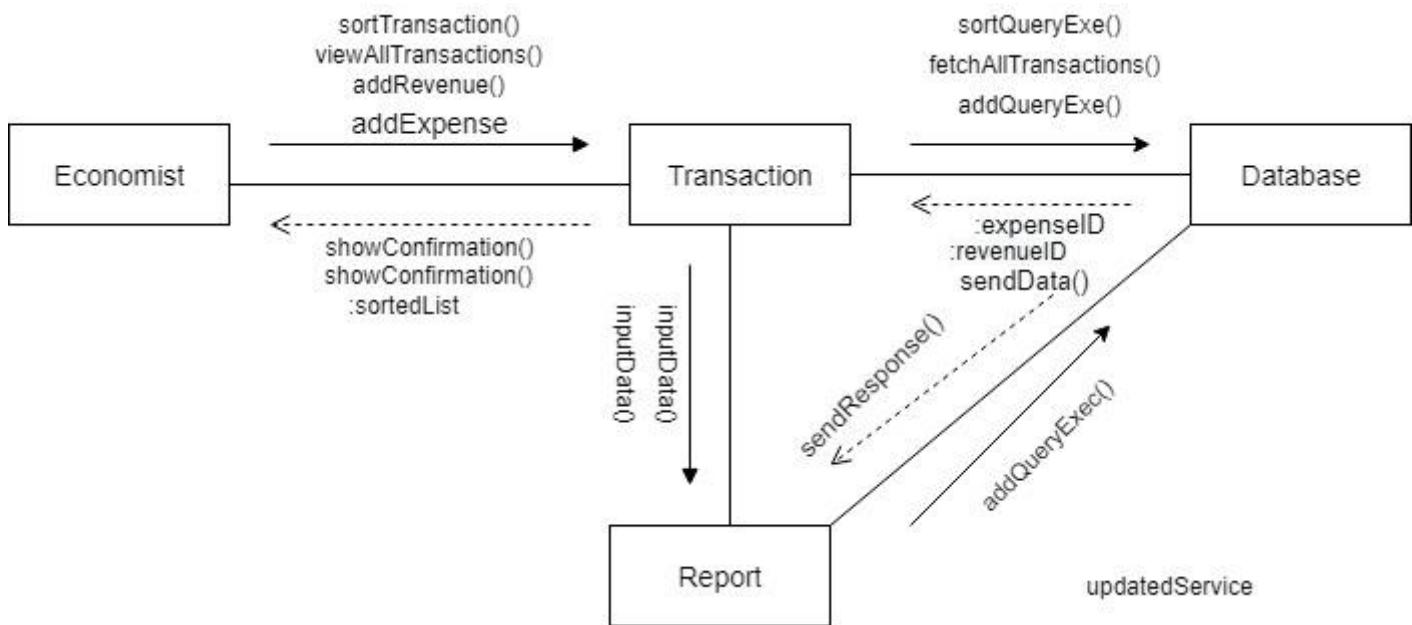
Col_2.4-2.1-3.1Emp-JobApp



Col_3.2.3-6.3_ManageDR

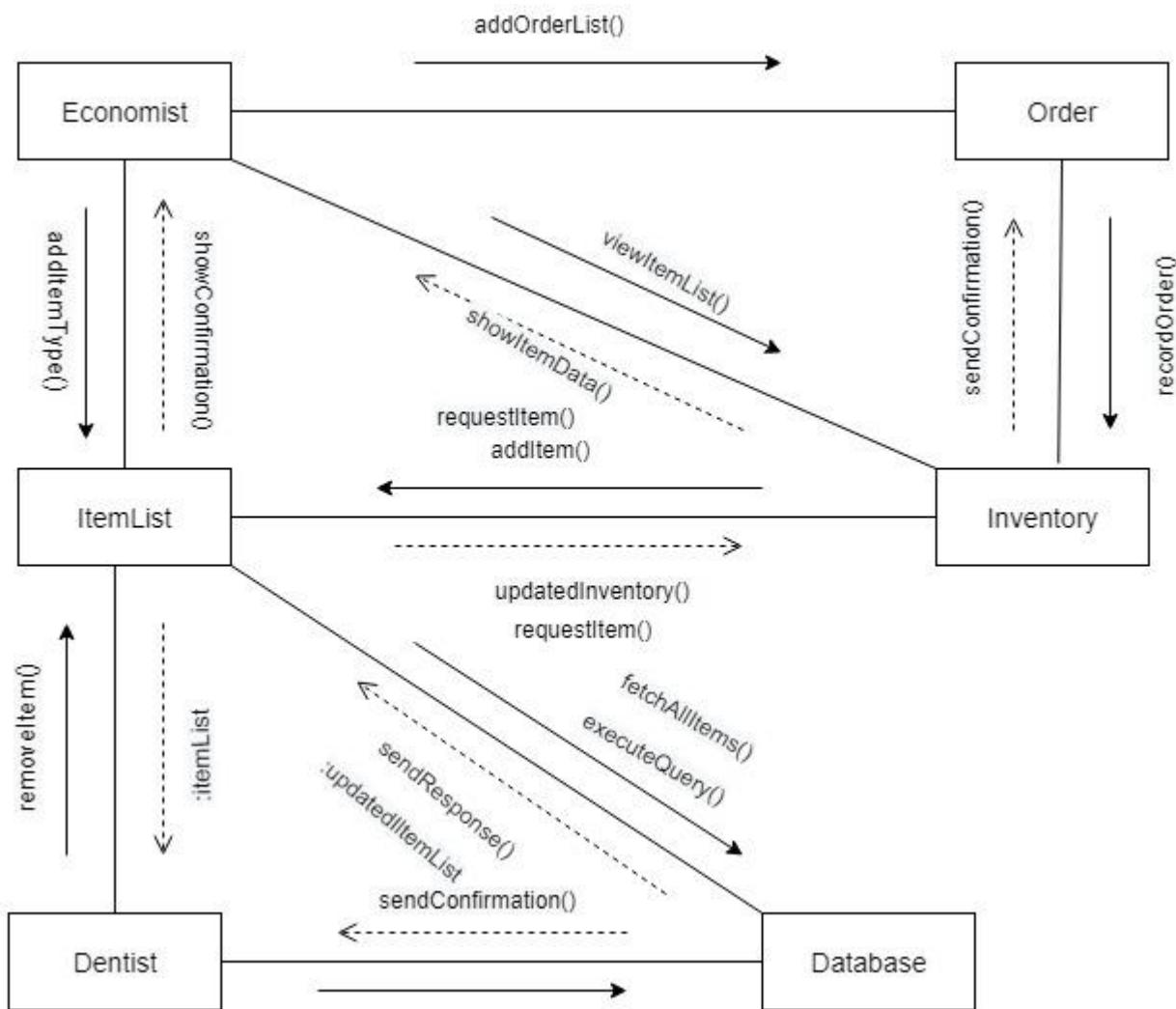


Col_7.2.4-7.1.1-7.1.4ManageTransaction



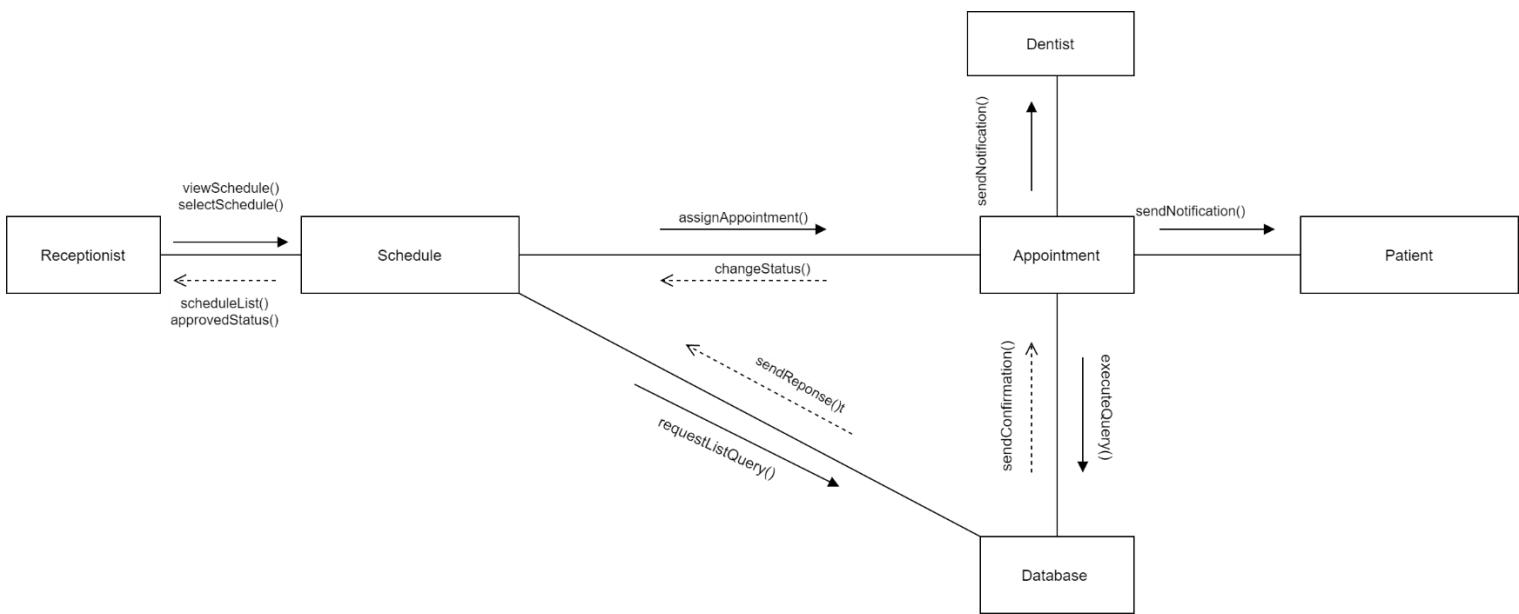
[E-DENT] Requirements Specification

Col_7.3.7-6.4-7.3.1ManageInventory

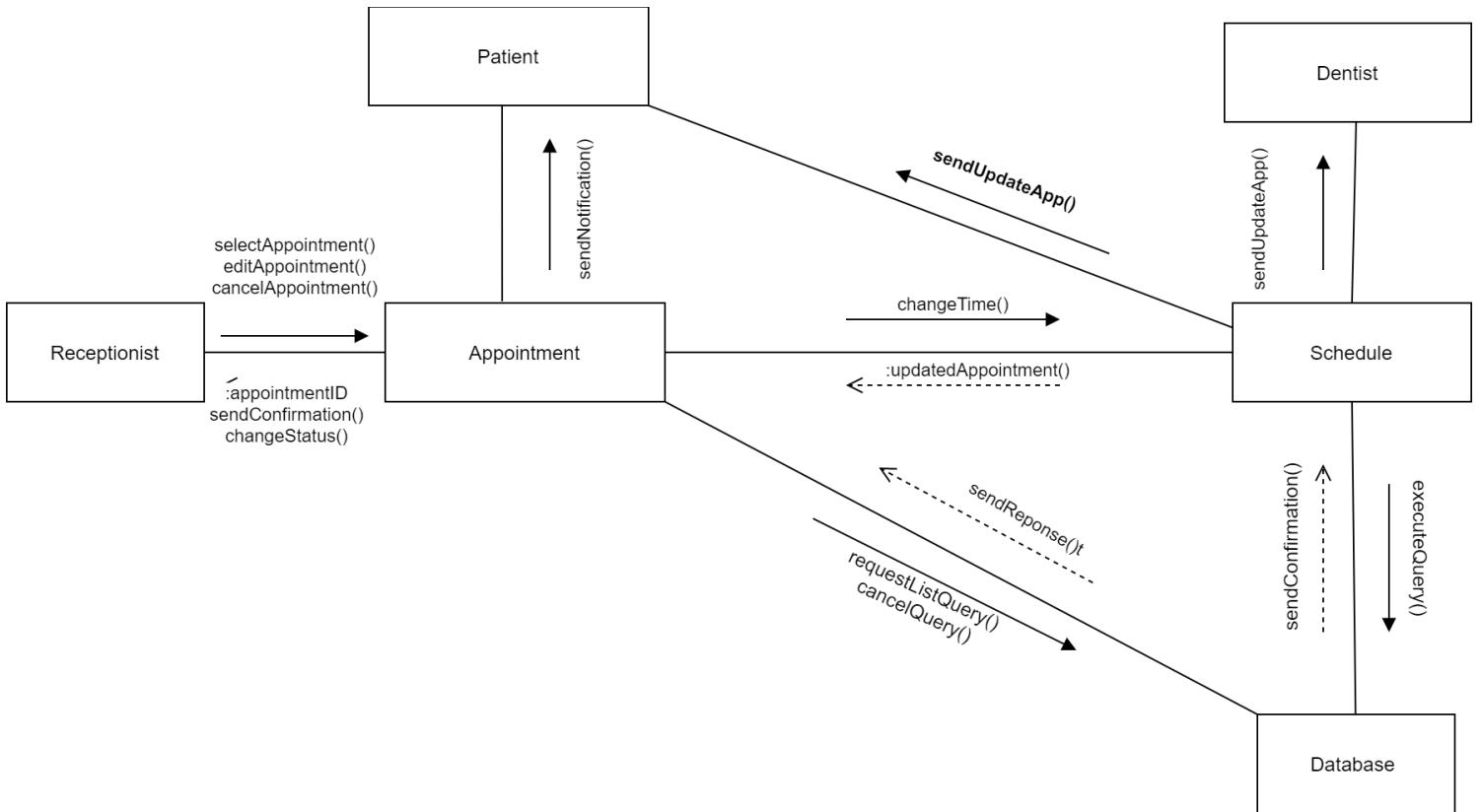


[E-DENT] Requirements Specification

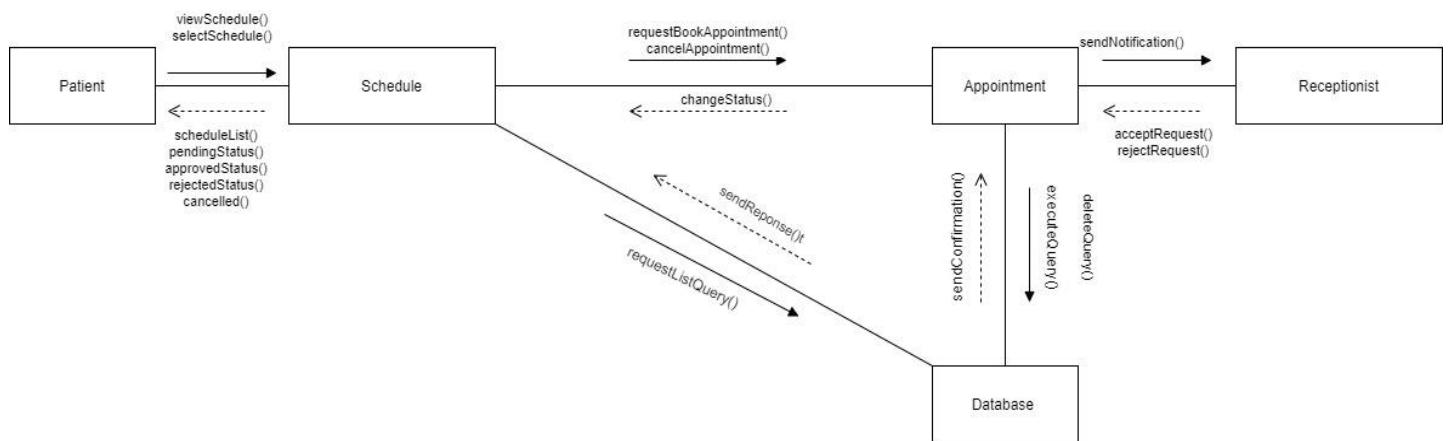
Col_5.2-5.1.2_AssignAppointment



Col_5.1.1-5.1.3-5.2>Edit Appointment

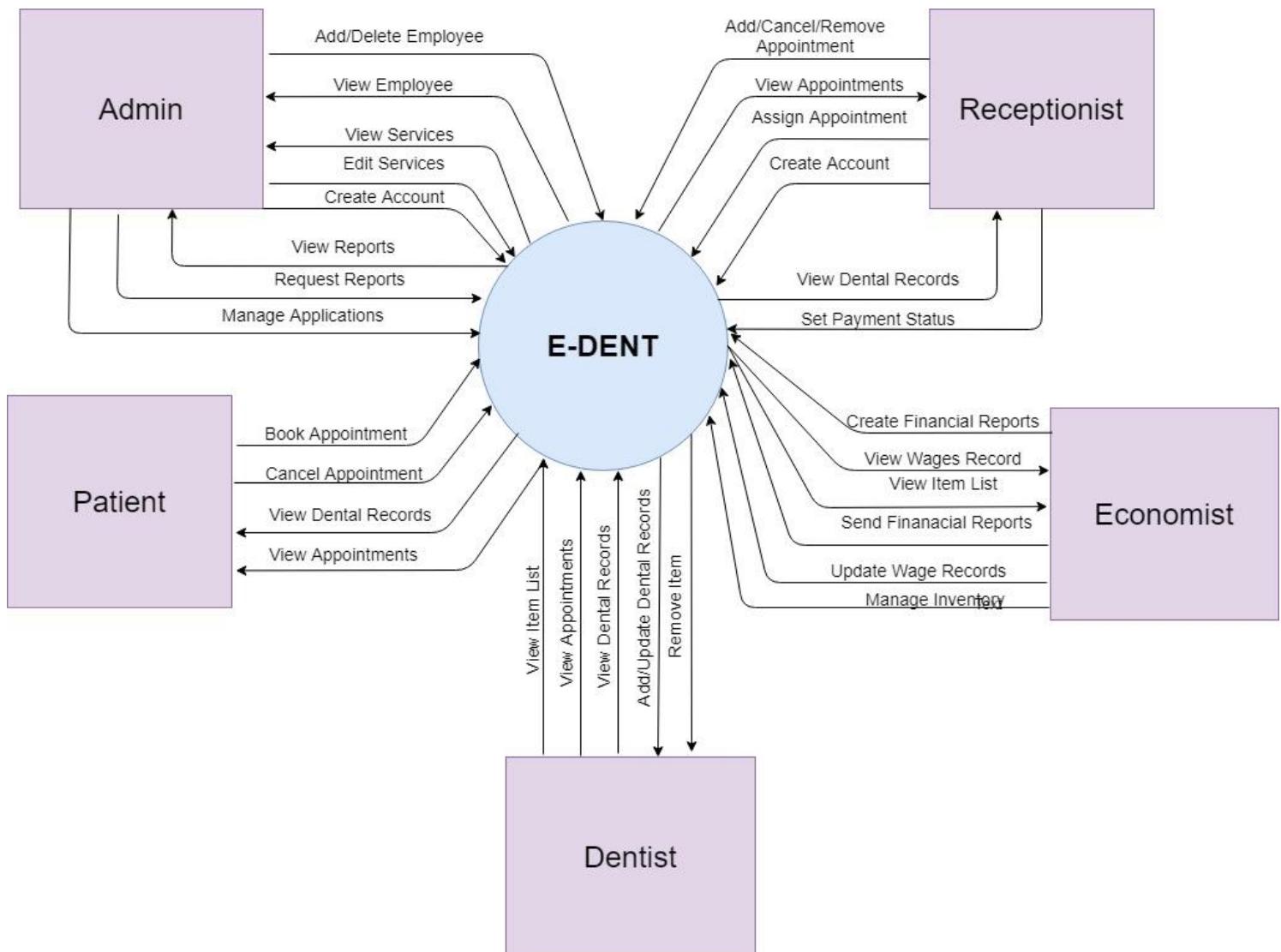


Col_4.2.1-4.2.2-5.1.2_Book Appointment

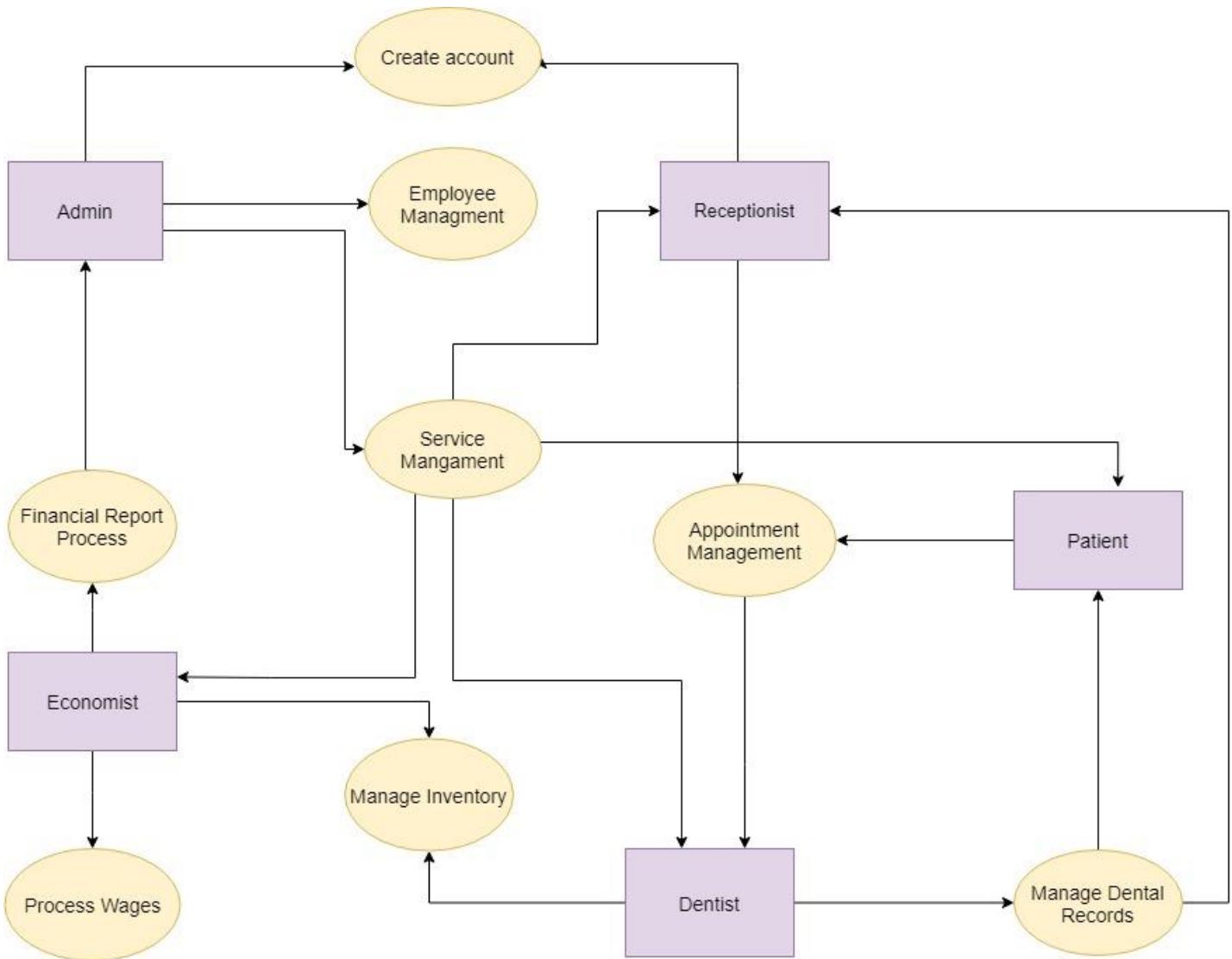


Data Flow Diagrams

Level 0

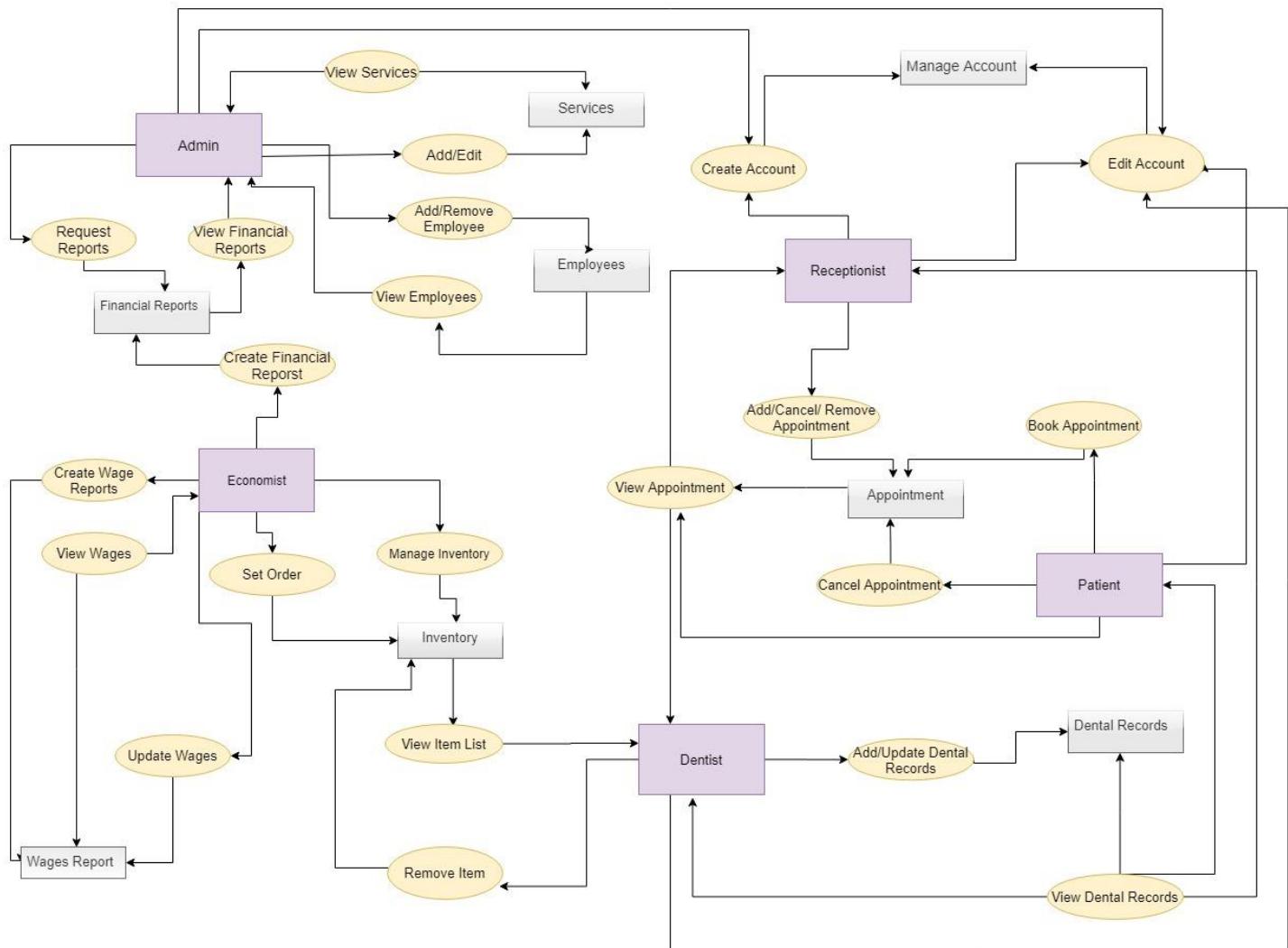


Level 1



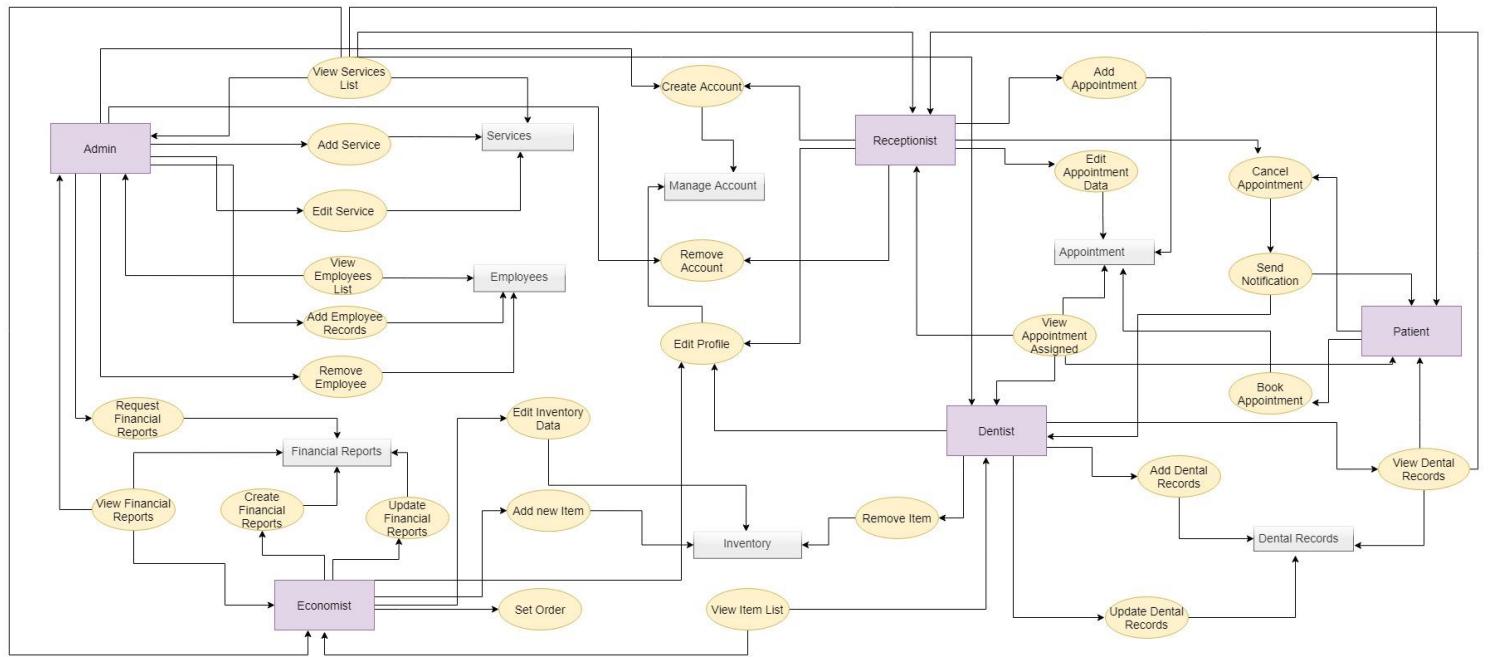
[E-DENT] Requirements Specification

Level 2

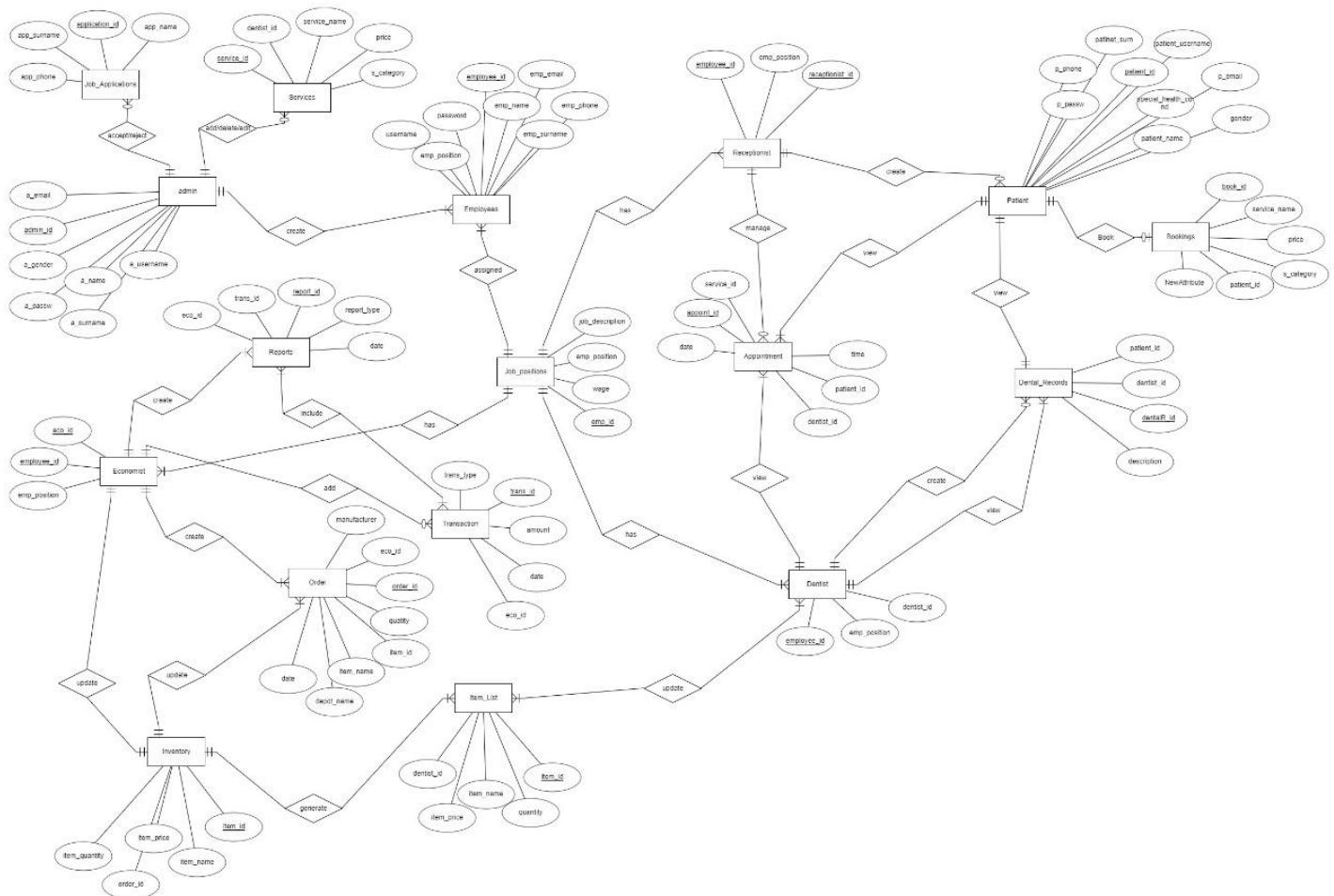


[E-DENT] Requirements Specification

Level 3



ERD

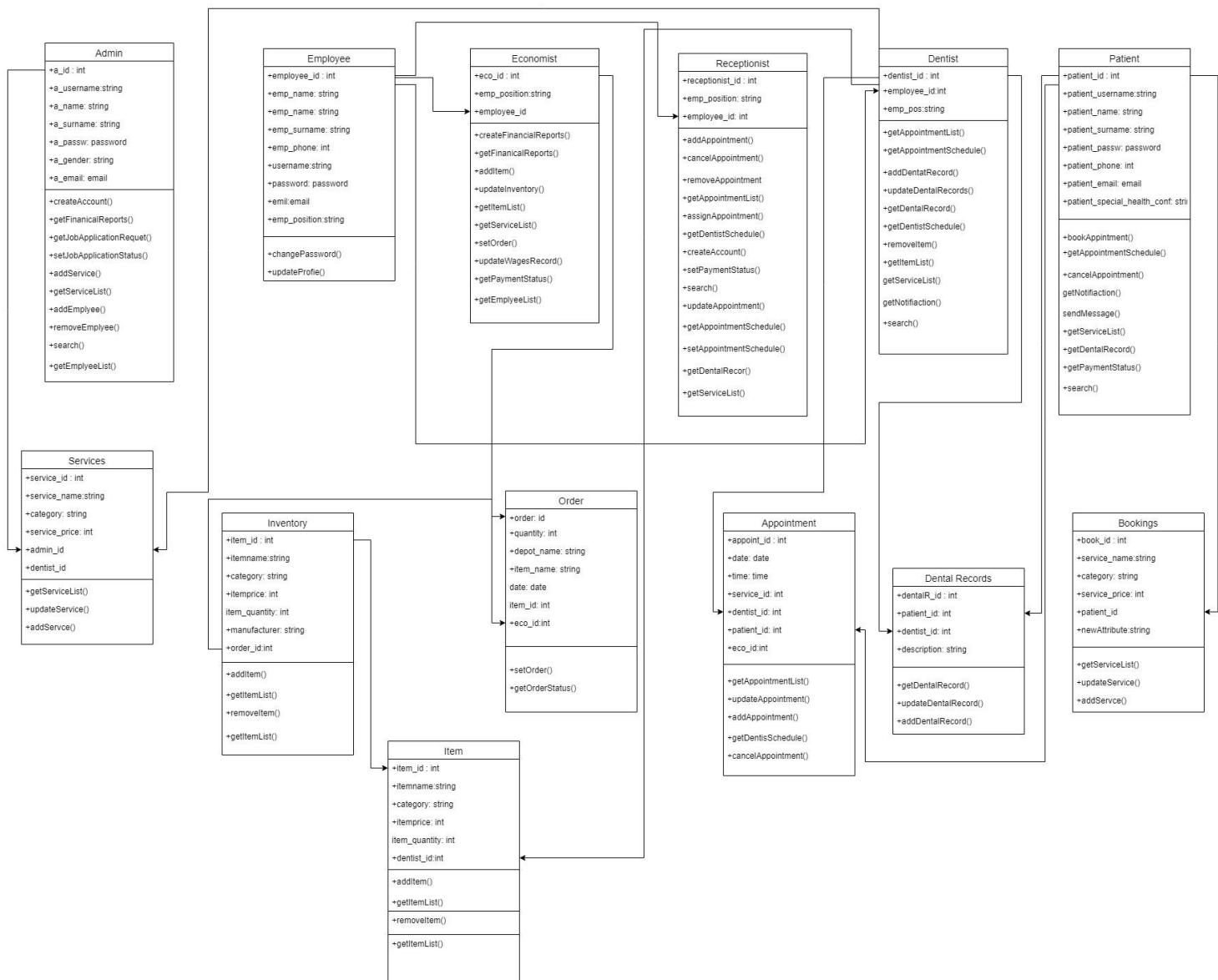


[E-DENT] Requirements Specification

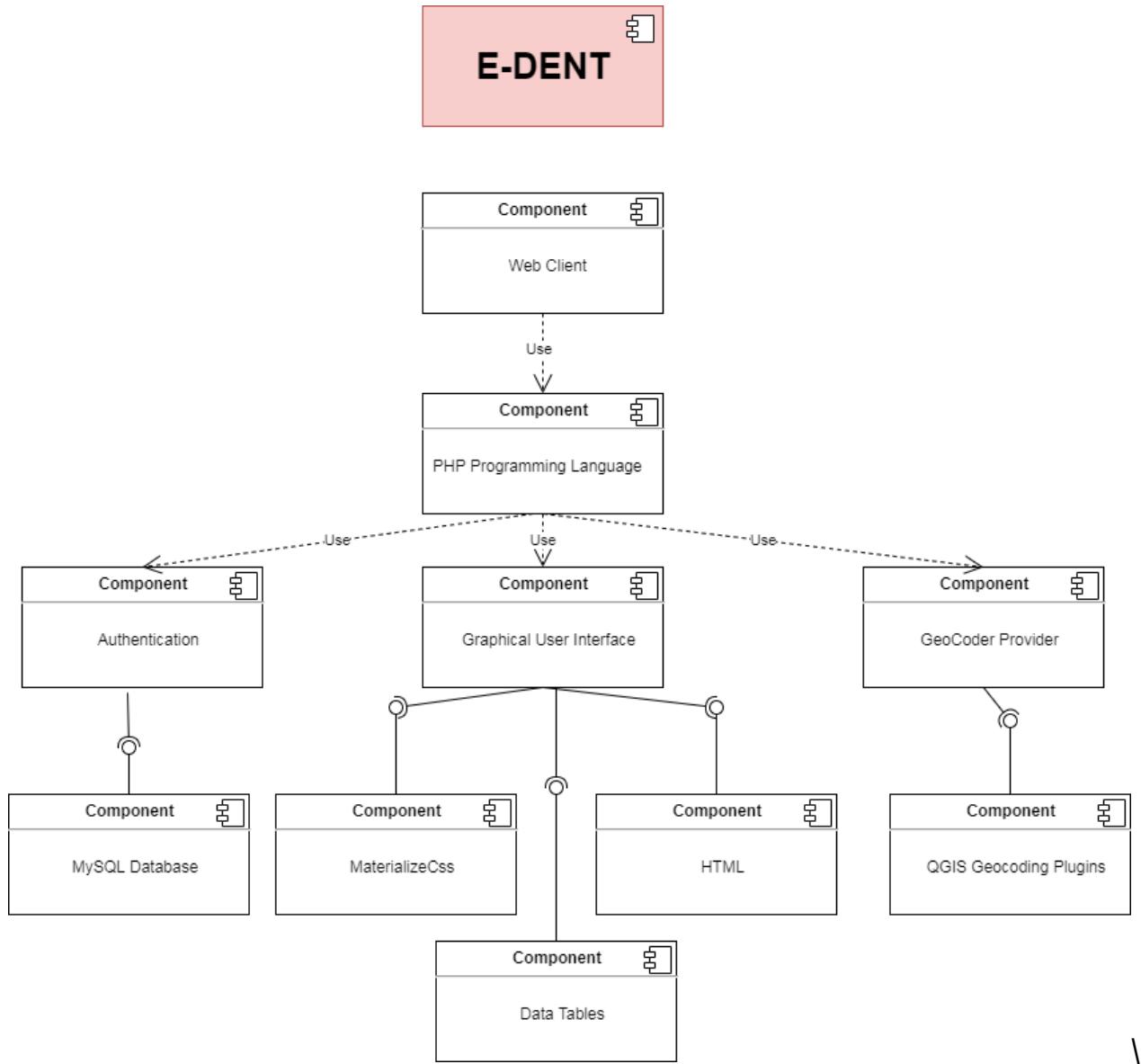
Classes

<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Admin</td></tr> <tr><td>+a_id : int</td><td></td></tr> <tr><td>+a_username string</td><td></td></tr> <tr><td>+a_name: string</td><td></td></tr> <tr><td>+a_surname: string</td><td></td></tr> <tr><td>+a_passw: password</td><td></td></tr> <tr><td>+a_gender: string</td><td></td></tr> <tr><td>+a_email: email</td><td></td></tr> <tr><td colspan="2">+createAccount()</td></tr> <tr><td colspan="2">+getFinancialReports()</td></tr> <tr><td colspan="2">+getJobApplicationRequest()</td></tr> <tr><td colspan="2">+setJobApplicationStatus()</td></tr> <tr><td colspan="2">+addService()</td></tr> <tr><td colspan="2">+getServiceList()</td></tr> <tr><td colspan="2">+addEmployee()</td></tr> <tr><td colspan="2">+removeEmployee()</td></tr> <tr><td colspan="2">+search()</td></tr> <tr><td colspan="2">+getEmployeeList()</td></tr> </table>	Admin		+a_id : int		+a_username string		+a_name: string		+a_surname: string		+a_passw: password		+a_gender: string		+a_email: email		+createAccount()		+getFinancialReports()		+getJobApplicationRequest()		+setJobApplicationStatus()		+addService()		+getServiceList()		+addEmployee()		+removeEmployee()		+search()		+getEmployeeList()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Employee</td></tr> <tr><td>+employee_id : int</td><td></td></tr> <tr><td>+emp_name: string</td><td></td></tr> <tr><td>+emp_surname: string</td><td></td></tr> <tr><td>+emp_phone: int</td><td></td></tr> <tr><td>+username: string</td><td></td></tr> <tr><td>+password: password</td><td></td></tr> <tr><td>+email: email</td><td></td></tr> <tr><td>+emp_position: string</td><td></td></tr> <tr><td colspan="2">+changePassword()</td></tr> <tr><td colspan="2">+updateProfile()</td></tr> </table>	Employee		+employee_id : int		+emp_name: string		+emp_surname: string		+emp_phone: int		+username: string		+password: password		+email: email		+emp_position: string		+changePassword()		+updateProfile()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Economist</td></tr> <tr><td>+eco_id : int</td><td></td></tr> <tr><td>+emp_position: string</td><td></td></tr> <tr><td>+employee_id</td><td></td></tr> <tr><td colspan="2">+createFinancialReports()</td></tr> <tr><td colspan="2">+getFinancialReports()</td></tr> <tr><td colspan="2">+addItem()</td></tr> <tr><td colspan="2">+updateInventory()</td></tr> <tr><td colspan="2">+getItemList()</td></tr> <tr><td colspan="2">+getServiceList()</td></tr> <tr><td colspan="2">+setOrder()</td></tr> <tr><td colspan="2">+updateWagesRecord()</td></tr> <tr><td colspan="2">+getPaymentStatus()</td></tr> <tr><td colspan="2">+getEmployeeList()</td></tr> </table>	Economist		+eco_id : int		+emp_position: string		+employee_id		+createFinancialReports()		+getFinancialReports()		+addItem()		+updateInventory()		+getItemList()		+getServiceList()		+setOrder()		+updateWagesRecord()		+getPaymentStatus()		+getEmployeeList()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Receptionist</td></tr> <tr><td>+receptionist_id : int</td><td></td></tr> <tr><td>+emp_position: string</td><td></td></tr> <tr><td>+employee_id: int</td><td></td></tr> <tr><td colspan="2">+addAppointment()</td></tr> <tr><td colspan="2">+cancelAppointment()</td></tr> <tr><td colspan="2">+removeAppointment</td></tr> <tr><td colspan="2">+getAppointmentList()</td></tr> <tr><td colspan="2">+assignAppointment()</td></tr> <tr><td colspan="2">+getDentistSchedule()</td></tr> <tr><td colspan="2">+createAccount()</td></tr> <tr><td colspan="2">+setPaymentStatus()</td></tr> <tr><td colspan="2">+search()</td></tr> <tr><td colspan="2">+updateAppointment()</td></tr> <tr><td colspan="2">+getAppointmentSchedule()</td></tr> <tr><td colspan="2">+setAppointmentSchedule()</td></tr> <tr><td colspan="2">+getDentalRecord()</td></tr> <tr><td colspan="2">+getServiceList()</td></tr> </table>	Receptionist		+receptionist_id : int		+emp_position: string		+employee_id: int		+addAppointment()		+cancelAppointment()		+removeAppointment		+getAppointmentList()		+assignAppointment()		+getDentistSchedule()		+createAccount()		+setPaymentStatus()		+search()		+updateAppointment()		+getAppointmentSchedule()		+setAppointmentSchedule()		+getDentalRecord()		+getServiceList()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Dentist</td></tr> <tr><td>+dentist_id : int</td><td></td></tr> <tr><td>+employee_id: int</td><td></td></tr> <tr><td>+emp_pos: string</td><td></td></tr> <tr><td colspan="2">+getAppointmentList()</td></tr> <tr><td colspan="2">+getAppointmentSchedule()</td></tr> <tr><td colspan="2">+addDentalRecord()</td></tr> <tr><td colspan="2">+updateDentalRecords()</td></tr> <tr><td colspan="2">+getDentalRecord()</td></tr> <tr><td colspan="2">+getDentistSchedule()</td></tr> <tr><td colspan="2">+removeItem()</td></tr> <tr><td colspan="2">+getItemList()</td></tr> <tr><td colspan="2">+getServiceList()</td></tr> <tr><td colspan="2">+getNotification()</td></tr> <tr><td colspan="2">+search()</td></tr> </table>	Dentist		+dentist_id : int		+employee_id: int		+emp_pos: string		+getAppointmentList()		+getAppointmentSchedule()		+addDentalRecord()		+updateDentalRecords()		+getDentalRecord()		+getDentistSchedule()		+removeItem()		+getItemList()		+getServiceList()		+getNotification()		+search()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Patient</td></tr> <tr><td>+patient_id : int</td><td></td></tr> <tr><td>+patient_username: string</td><td></td></tr> <tr><td>+patient_name: string</td><td></td></tr> <tr><td>+patient_surname: string</td><td></td></tr> <tr><td>+patient_passw: password</td><td></td></tr> <tr><td>+patient_phone: int</td><td></td></tr> <tr><td>+patient_email: email</td><td></td></tr> <tr><td>+patient_special_health_conf: string</td><td></td></tr> <tr><td colspan="2">+bookAppointment()</td></tr> <tr><td colspan="2">+getAppointmentSchedule()</td></tr> <tr><td colspan="2">+cancelAppointment()</td></tr> <tr><td colspan="2">getNotification()</td></tr> <tr><td colspan="2">sendMessage()</td></tr> <tr><td colspan="2">+getServiceList()</td></tr> <tr><td colspan="2">+getDentalRecord()</td></tr> <tr><td colspan="2">+getPaymentStatus()</td></tr> <tr><td colspan="2">+search()</td></tr> </table>	Patient		+patient_id : int		+patient_username: string		+patient_name: string		+patient_surname: string		+patient_passw: password		+patient_phone: int		+patient_email: email		+patient_special_health_conf: string		+bookAppointment()		+getAppointmentSchedule()		+cancelAppointment()		getNotification()		sendMessage()		+getServiceList()		+getDentalRecord()		+getPaymentStatus()		+search()	
Admin																																																																																																																																																																																																	
+a_id : int																																																																																																																																																																																																	
+a_username string																																																																																																																																																																																																	
+a_name: string																																																																																																																																																																																																	
+a_surname: string																																																																																																																																																																																																	
+a_passw: password																																																																																																																																																																																																	
+a_gender: string																																																																																																																																																																																																	
+a_email: email																																																																																																																																																																																																	
+createAccount()																																																																																																																																																																																																	
+getFinancialReports()																																																																																																																																																																																																	
+getJobApplicationRequest()																																																																																																																																																																																																	
+setJobApplicationStatus()																																																																																																																																																																																																	
+addService()																																																																																																																																																																																																	
+getServiceList()																																																																																																																																																																																																	
+addEmployee()																																																																																																																																																																																																	
+removeEmployee()																																																																																																																																																																																																	
+search()																																																																																																																																																																																																	
+getEmployeeList()																																																																																																																																																																																																	
Employee																																																																																																																																																																																																	
+employee_id : int																																																																																																																																																																																																	
+emp_name: string																																																																																																																																																																																																	
+emp_surname: string																																																																																																																																																																																																	
+emp_phone: int																																																																																																																																																																																																	
+username: string																																																																																																																																																																																																	
+password: password																																																																																																																																																																																																	
+email: email																																																																																																																																																																																																	
+emp_position: string																																																																																																																																																																																																	
+changePassword()																																																																																																																																																																																																	
+updateProfile()																																																																																																																																																																																																	
Economist																																																																																																																																																																																																	
+eco_id : int																																																																																																																																																																																																	
+emp_position: string																																																																																																																																																																																																	
+employee_id																																																																																																																																																																																																	
+createFinancialReports()																																																																																																																																																																																																	
+getFinancialReports()																																																																																																																																																																																																	
+addItem()																																																																																																																																																																																																	
+updateInventory()																																																																																																																																																																																																	
+getItemList()																																																																																																																																																																																																	
+getServiceList()																																																																																																																																																																																																	
+setOrder()																																																																																																																																																																																																	
+updateWagesRecord()																																																																																																																																																																																																	
+getPaymentStatus()																																																																																																																																																																																																	
+getEmployeeList()																																																																																																																																																																																																	
Receptionist																																																																																																																																																																																																	
+receptionist_id : int																																																																																																																																																																																																	
+emp_position: string																																																																																																																																																																																																	
+employee_id: int																																																																																																																																																																																																	
+addAppointment()																																																																																																																																																																																																	
+cancelAppointment()																																																																																																																																																																																																	
+removeAppointment																																																																																																																																																																																																	
+getAppointmentList()																																																																																																																																																																																																	
+assignAppointment()																																																																																																																																																																																																	
+getDentistSchedule()																																																																																																																																																																																																	
+createAccount()																																																																																																																																																																																																	
+setPaymentStatus()																																																																																																																																																																																																	
+search()																																																																																																																																																																																																	
+updateAppointment()																																																																																																																																																																																																	
+getAppointmentSchedule()																																																																																																																																																																																																	
+setAppointmentSchedule()																																																																																																																																																																																																	
+getDentalRecord()																																																																																																																																																																																																	
+getServiceList()																																																																																																																																																																																																	
Dentist																																																																																																																																																																																																	
+dentist_id : int																																																																																																																																																																																																	
+employee_id: int																																																																																																																																																																																																	
+emp_pos: string																																																																																																																																																																																																	
+getAppointmentList()																																																																																																																																																																																																	
+getAppointmentSchedule()																																																																																																																																																																																																	
+addDentalRecord()																																																																																																																																																																																																	
+updateDentalRecords()																																																																																																																																																																																																	
+getDentalRecord()																																																																																																																																																																																																	
+getDentistSchedule()																																																																																																																																																																																																	
+removeItem()																																																																																																																																																																																																	
+getItemList()																																																																																																																																																																																																	
+getServiceList()																																																																																																																																																																																																	
+getNotification()																																																																																																																																																																																																	
+search()																																																																																																																																																																																																	
Patient																																																																																																																																																																																																	
+patient_id : int																																																																																																																																																																																																	
+patient_username: string																																																																																																																																																																																																	
+patient_name: string																																																																																																																																																																																																	
+patient_surname: string																																																																																																																																																																																																	
+patient_passw: password																																																																																																																																																																																																	
+patient_phone: int																																																																																																																																																																																																	
+patient_email: email																																																																																																																																																																																																	
+patient_special_health_conf: string																																																																																																																																																																																																	
+bookAppointment()																																																																																																																																																																																																	
+getAppointmentSchedule()																																																																																																																																																																																																	
+cancelAppointment()																																																																																																																																																																																																	
getNotification()																																																																																																																																																																																																	
sendMessage()																																																																																																																																																																																																	
+getServiceList()																																																																																																																																																																																																	
+getDentalRecord()																																																																																																																																																																																																	
+getPaymentStatus()																																																																																																																																																																																																	
+search()																																																																																																																																																																																																	
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Services</td></tr> <tr><td>+service_id : int</td><td></td></tr> <tr><td>+service_name: string</td><td></td></tr> <tr><td>+category: string</td><td></td></tr> <tr><td>+service_price: int</td><td></td></tr> <tr><td>+dentist_id</td><td></td></tr> <tr><td colspan="2">+getServiceList()</td></tr> <tr><td colspan="2">+updateService()</td></tr> <tr><td colspan="2">+addService()</td></tr> </table>	Services		+service_id : int		+service_name: string		+category: string		+service_price: int		+dentist_id		+getServiceList()		+updateService()		+addService()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Inventory</td></tr> <tr><td>+item_id : int</td><td></td></tr> <tr><td>+itemname: string</td><td></td></tr> <tr><td>+category: string</td><td></td></tr> <tr><td>+itemprice: int</td><td></td></tr> <tr><td>item_quantity: int</td><td></td></tr> <tr><td>+manufacturer: string</td><td></td></tr> <tr><td>+order_id: int</td><td></td></tr> <tr><td colspan="2">+addItem()</td></tr> <tr><td colspan="2">+getItemList()</td></tr> <tr><td colspan="2">+removeItem()</td></tr> <tr><td colspan="2">+getItemList()</td></tr> </table>	Inventory		+item_id : int		+itemname: string		+category: string		+itemprice: int		item_quantity: int		+manufacturer: string		+order_id: int		+addItem()		+getItemList()		+removeItem()		+getItemList()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Item</td></tr> <tr><td>+item_id : int</td><td></td></tr> <tr><td>+itemname: string</td><td></td></tr> <tr><td>+category: string</td><td></td></tr> <tr><td>+itemprice: int</td><td></td></tr> <tr><td>item_quantity: int</td><td></td></tr> <tr><td>+order_id: int</td><td></td></tr> <tr><td colspan="2">+addItem()</td></tr> <tr><td colspan="2">+getItemList()</td></tr> <tr><td colspan="2">+removeItem()</td></tr> <tr><td colspan="2">+getItemList()</td></tr> </table>	Item		+item_id : int		+itemname: string		+category: string		+itemprice: int		item_quantity: int		+order_id: int		+addItem()		+getItemList()		+removeItem()		+getItemList()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Order</td></tr> <tr><td>+order_id</td><td></td></tr> <tr><td>+quantity: int</td><td></td></tr> <tr><td>+depot_name: string</td><td></td></tr> <tr><td>+item_name: string</td><td></td></tr> <tr><td>date: date</td><td></td></tr> <tr><td>item_id: int</td><td></td></tr> <tr><td>+eco_id: int</td><td></td></tr> <tr><td colspan="2">+setOrder()</td></tr> <tr><td colspan="2">+getOrderStatus()</td></tr> </table>	Order		+order_id		+quantity: int		+depot_name: string		+item_name: string		date: date		item_id: int		+eco_id: int		+setOrder()		+getOrderStatus()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Appointment</td></tr> <tr><td>+appoint_id : int</td><td></td></tr> <tr><td>+date: date</td><td></td></tr> <tr><td>+time: time</td><td></td></tr> <tr><td>+service_id: int</td><td></td></tr> <tr><td>+dentist_id: int</td><td></td></tr> <tr><td>+patient_id: int</td><td></td></tr> <tr><td>+eco_id: int</td><td></td></tr> <tr><td colspan="2">+getAppointmentList()</td></tr> <tr><td colspan="2">+updateAppointment()</td></tr> <tr><td colspan="2">+addAppointment()</td></tr> <tr><td colspan="2">+getDentistSchedule()</td></tr> <tr><td colspan="2">+cancelAppointment()</td></tr> </table>	Appointment		+appoint_id : int		+date: date		+time: time		+service_id: int		+dentist_id: int		+patient_id: int		+eco_id: int		+getAppointmentList()		+updateAppointment()		+addAppointment()		+getDentistSchedule()		+cancelAppointment()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Dental Records</td></tr> <tr><td>+dentalR_id : int</td><td></td></tr> <tr><td>+patient_id: int</td><td></td></tr> <tr><td>+dentist_id: int</td><td></td></tr> <tr><td>+description: string</td><td></td></tr> <tr><td colspan="2">+getServiceList()</td></tr> <tr><td colspan="2">+updateDentalRecord()</td></tr> <tr><td colspan="2">+addDentalRecord()</td></tr> </table>	Dental Records		+dentalR_id : int		+patient_id: int		+dentist_id: int		+description: string		+getServiceList()		+updateDentalRecord()		+addDentalRecord()		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2">Bookings</td></tr> <tr><td>+book_id : int</td><td></td></tr> <tr><td>+service_name: string</td><td></td></tr> <tr><td>+category: string</td><td></td></tr> <tr><td>+service_price: int</td><td></td></tr> <tr><td>+patient_id</td><td></td></tr> <tr><td>+newAttribute: string</td><td></td></tr> <tr><td colspan="2">+getServiceList()</td></tr> <tr><td colspan="2">+updateService()</td></tr> <tr><td colspan="2">+addService()</td></tr> </table>	Bookings		+book_id : int		+service_name: string		+category: string		+service_price: int		+patient_id		+newAttribute: string		+getServiceList()		+updateService()		+addService()																																										
Services																																																																																																																																																																																																	
+service_id : int																																																																																																																																																																																																	
+service_name: string																																																																																																																																																																																																	
+category: string																																																																																																																																																																																																	
+service_price: int																																																																																																																																																																																																	
+dentist_id																																																																																																																																																																																																	
+getServiceList()																																																																																																																																																																																																	
+updateService()																																																																																																																																																																																																	
+addService()																																																																																																																																																																																																	
Inventory																																																																																																																																																																																																	
+item_id : int																																																																																																																																																																																																	
+itemname: string																																																																																																																																																																																																	
+category: string																																																																																																																																																																																																	
+itemprice: int																																																																																																																																																																																																	
item_quantity: int																																																																																																																																																																																																	
+manufacturer: string																																																																																																																																																																																																	
+order_id: int																																																																																																																																																																																																	
+addItem()																																																																																																																																																																																																	
+getItemList()																																																																																																																																																																																																	
+removeItem()																																																																																																																																																																																																	
+getItemList()																																																																																																																																																																																																	
Item																																																																																																																																																																																																	
+item_id : int																																																																																																																																																																																																	
+itemname: string																																																																																																																																																																																																	
+category: string																																																																																																																																																																																																	
+itemprice: int																																																																																																																																																																																																	
item_quantity: int																																																																																																																																																																																																	
+order_id: int																																																																																																																																																																																																	
+addItem()																																																																																																																																																																																																	
+getItemList()																																																																																																																																																																																																	
+removeItem()																																																																																																																																																																																																	
+getItemList()																																																																																																																																																																																																	
Order																																																																																																																																																																																																	
+order_id																																																																																																																																																																																																	
+quantity: int																																																																																																																																																																																																	
+depot_name: string																																																																																																																																																																																																	
+item_name: string																																																																																																																																																																																																	
date: date																																																																																																																																																																																																	
item_id: int																																																																																																																																																																																																	
+eco_id: int																																																																																																																																																																																																	
+setOrder()																																																																																																																																																																																																	
+getOrderStatus()																																																																																																																																																																																																	
Appointment																																																																																																																																																																																																	
+appoint_id : int																																																																																																																																																																																																	
+date: date																																																																																																																																																																																																	
+time: time																																																																																																																																																																																																	
+service_id: int																																																																																																																																																																																																	
+dentist_id: int																																																																																																																																																																																																	
+patient_id: int																																																																																																																																																																																																	
+eco_id: int																																																																																																																																																																																																	
+getAppointmentList()																																																																																																																																																																																																	
+updateAppointment()																																																																																																																																																																																																	
+addAppointment()																																																																																																																																																																																																	
+getDentistSchedule()																																																																																																																																																																																																	
+cancelAppointment()																																																																																																																																																																																																	
Dental Records																																																																																																																																																																																																	
+dentalR_id : int																																																																																																																																																																																																	
+patient_id: int																																																																																																																																																																																																	
+dentist_id: int																																																																																																																																																																																																	
+description: string																																																																																																																																																																																																	
+getServiceList()																																																																																																																																																																																																	
+updateDentalRecord()																																																																																																																																																																																																	
+addDentalRecord()																																																																																																																																																																																																	
Bookings																																																																																																																																																																																																	
+book_id : int																																																																																																																																																																																																	
+service_name: string																																																																																																																																																																																																	
+category: string																																																																																																																																																																																																	
+service_price: int																																																																																																																																																																																																	
+patient_id																																																																																																																																																																																																	
+newAttribute: string																																																																																																																																																																																																	
+getServiceList()																																																																																																																																																																																																	
+updateService()																																																																																																																																																																																																	
+addService()																																																																																																																																																																																																	

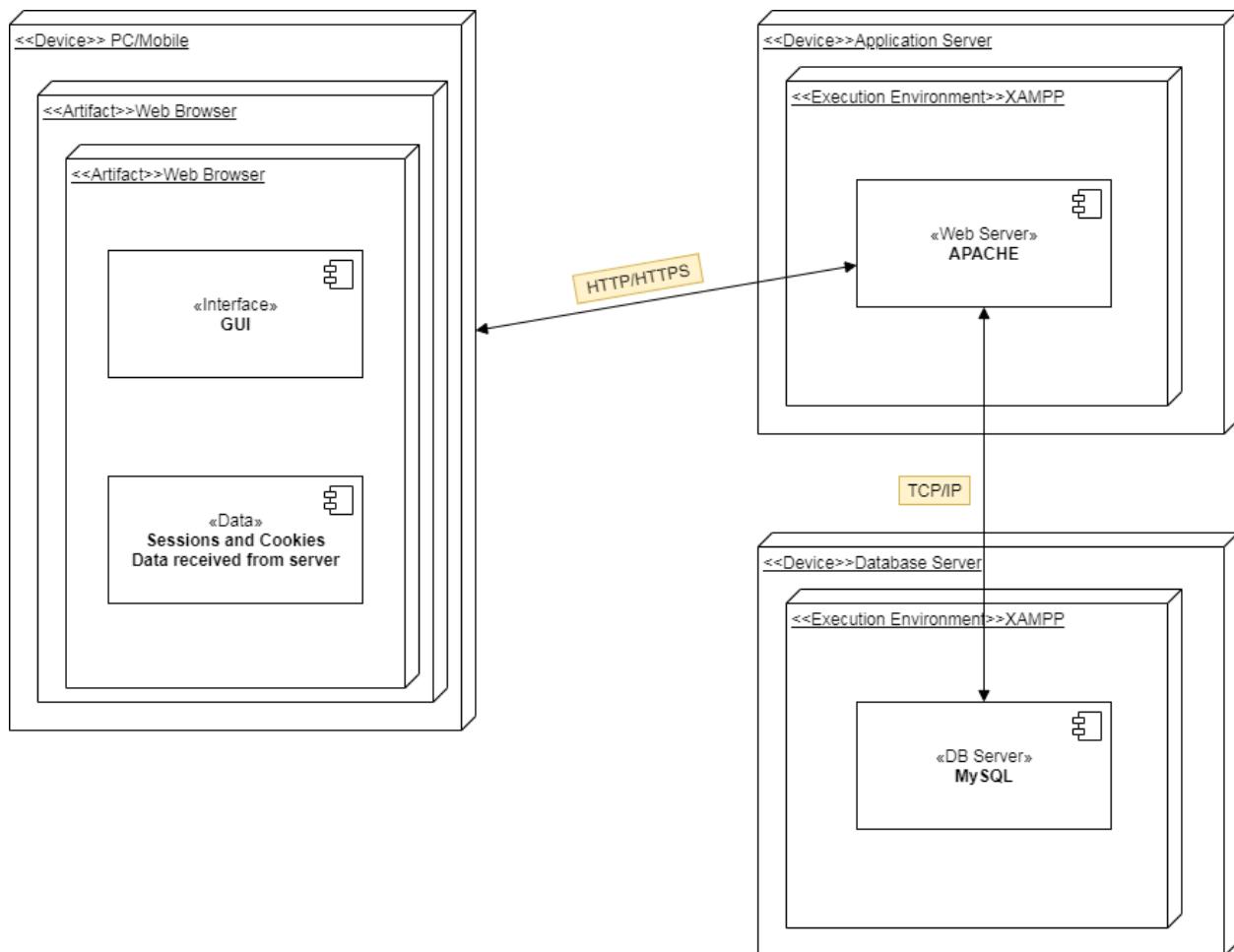
Class Diagram



COMPONENT DIAGRAM



DEPLOYMENT DIAGRAM



[E-DENT] Requirements Specification