

(Mantech Helpdesk)

Project Documentation



Team Members

Student Name	Student Number	Role
M Maryam Fahd Hajeb	Student1256886	Leader
A Abdu Khalid Al-hrazi	Student1256999	Member
H Hesham Mohammed Noman	Student1256882	Member
O Osama Nasser Al-koli	Student1256894	Member
S Mohammed Nabeel Shwabi	Student1256804	Member



Table of Content

1) Certificate of completion.

2) Acknowledgment.

3) Project synopsis:

- Problem statement.
- Proposed solution.

4) Project Analysis.

- Customer Requirements Specification.
- Hardware/ Software Requirements.
- Accomplished tasks.

5) Design:

- Data flow diagram.
- Flowcharts.
- Process diagram.
- Database design/ structure.

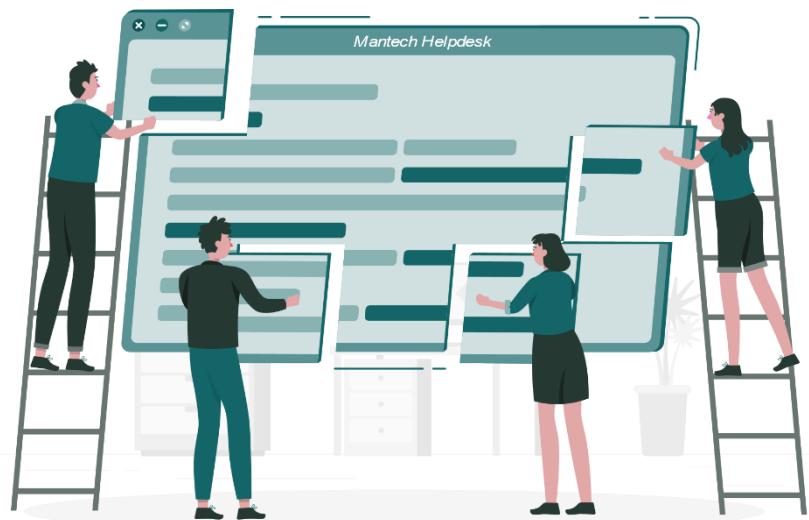
6) Screen shots.

7) Testing.

8) Source code with comments.

9) User guide.

10) Developer's guide.





CERTIFICATE

OF COMPLETION

— PRESENTED TO —

Abdu Khalid Al-Hrazi, Hesham Mohammed Noaman,
Maryam Fahd Hajeb, Mohammed Nabeel
Shawbi, Osama Nasser Al-koli

**Has successfully completed the developing and designing
“Mantech Helpdesk Website”**

Submitted to

AptechWorld Wide

Acknowledgment

While bringing out this thesis to its final form, we came across a number of people whose contributions in various ways helped through the completion of this project and they deserve special thanks. It is a pleasure to convey our gratitude to all of them.

First and foremost, we would thank Allah for being able to complete this project with success. Then we would like to express our deep sense of gratitude and indebtedness to Aptech Institute who gave us this opportunity to work on this project. We got to learn a lot from this project about how to work on real life scenarios and how to provide a more robust application to solve these scenarios. This project also helped us in improving our android programming language skills and techniques.

We specially acknowledge Dr. Qais Ali Al-Nuzali, principle of IT department, Eng. Fawaz Mohammad Hayel Othman and Dr. Hany Alghaithi, and all the staff of the IT department of Al-Nasser University for their valuable suggestions and guidance which have been very helpful in various stages of project completion.

Finally, we must also thank our parents and friends for the immense support and help during this project. Without their help, completing this project would have been very difficult.

Project synopsis

Problem statement

Mantech Limited wants to implement Customer Helpdesk Management for Technical Support. It is encountered that there are too many requests. The registration of these complaints/queries is manual which involves filling of complaint forms and maintaining records on paper.

For the past few months they have been facing problems with this process. Due to the problems they have felt the need for a change in their system. It has been observed that there are a lot of Transactions that are happening daily which if automated would help them to serve their staff better and faster.



Proposed Solution

We have recommended that they create a website where employees can submit complaints about any hardware or software issues with their machines.

All these complaints will be stored in a database, and the administrator will be able to view the daily registered complaints and assign a technician to resolve them.

Once the complaint or problem is resolved, the technician will need to enter the details on the website and close the complaint.

Employees can submit new complaints and check the status of their previous complaints on the website. Similarly, the administrator can also view the status of all complaints on the website and adjust their priority. The administrator has the ability to generate various types of reports.

Project Analysis

- Customer Requirements Specification:
 - ❖ Any Phone use Android System
 - ❖ 2GB or More and Ram 2 for performance better
 - ❖ The Phone System more than android 5.0 lollipop

- Hardware/ Software Requirements :

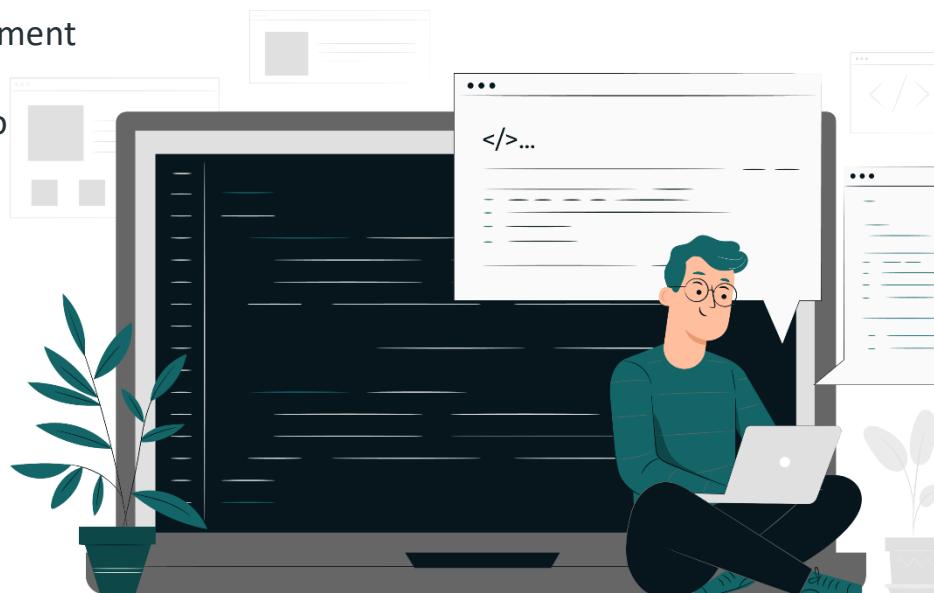
Hardware

- ❖ A minimum computer system that will help you access all the tools inthe courses is a Pentium 166 or better
- ❖ 64 Megabytes of RAM or better
- ❖ Windows 2000 (or higher if possible)
- ❖ Java Virtual Machine

Software

Use software as per your requirement

- ❖ Notepad/Java editor/visual studio
- ❖ j2sdk1.4.1_02 (or later).
- ❖ EJB Dev Kit
- ❖ Java enabled web server
- ❖ JSF



- Accomplished tasks:

M	Maryam Fahd Hajeb
A	Abdu Khalid Al-hrazi
H	Hesham Mohammed Noman
O	Osama Nasser Al-koli
S	Mohammed Nabeel Shwabi



N	Task	Prepared By	Start Date	End Date	Status
1	Review eProject and Planning	All Team			Done
2	Requirement Gathering	All Team			Done
3	UI Design	M			Done
4	Analysis	All Team			Done
5	Coding	M A H S			Done
6	Testing	M A H S			Done
7	Documentation	M A H S			Done

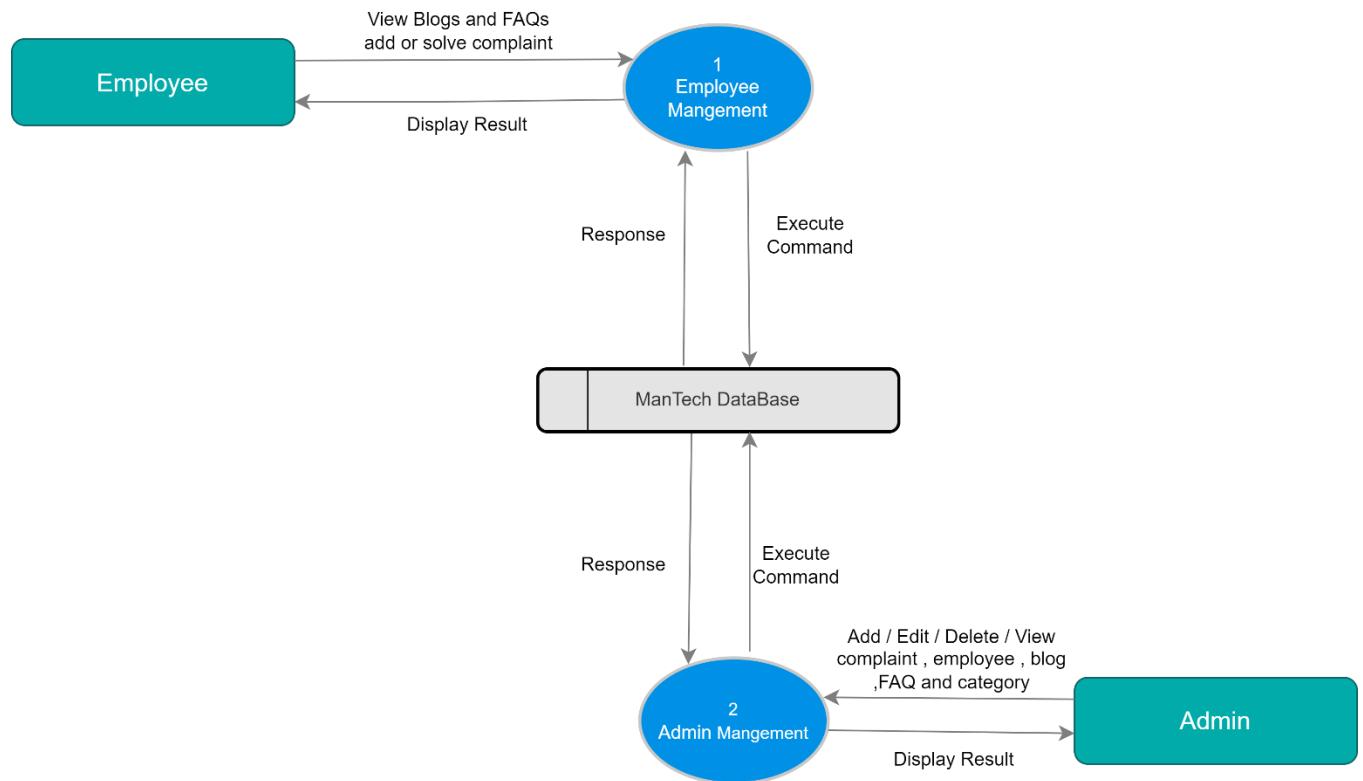
Project Design

Data flow diagram

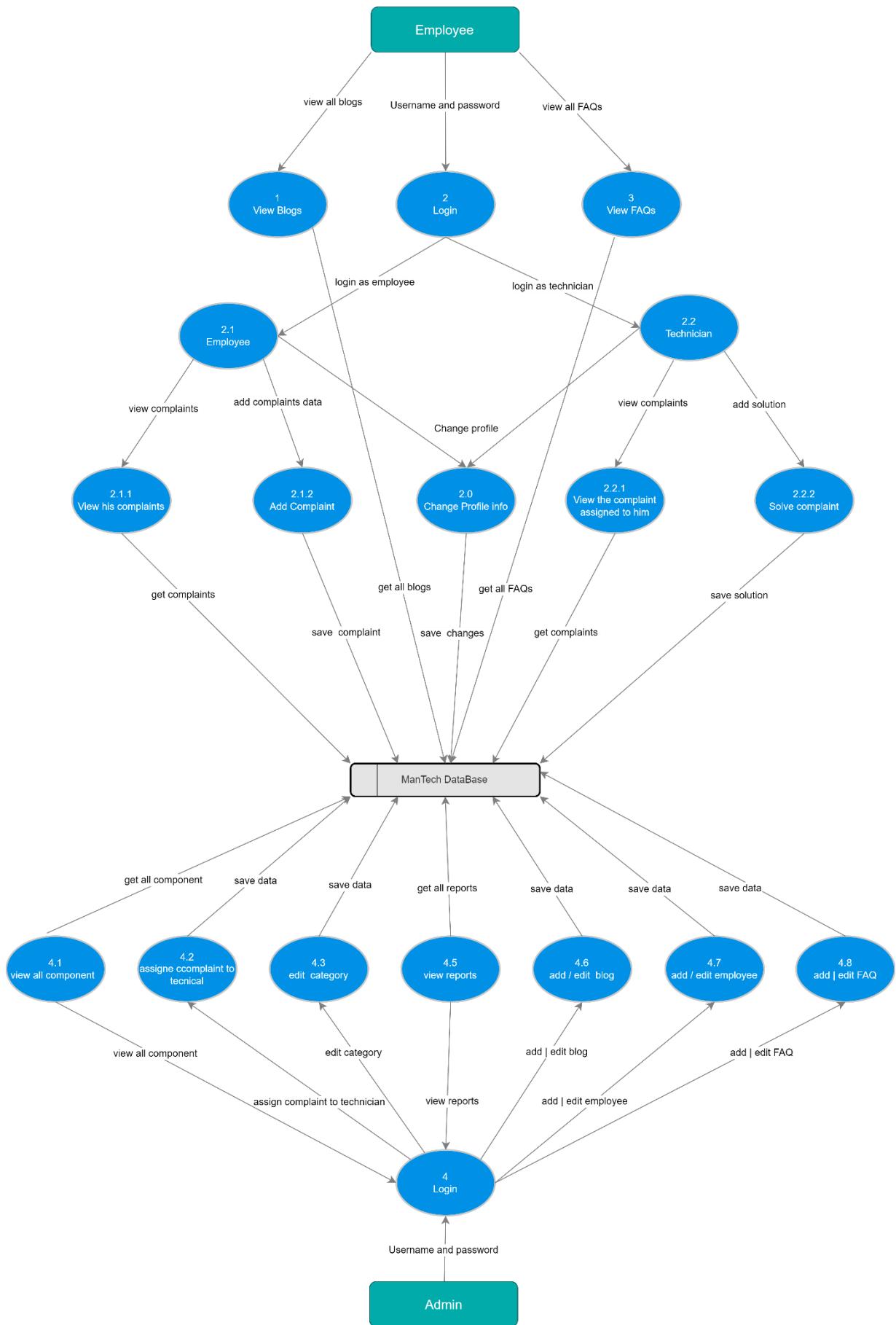
- Level 0 DFD:



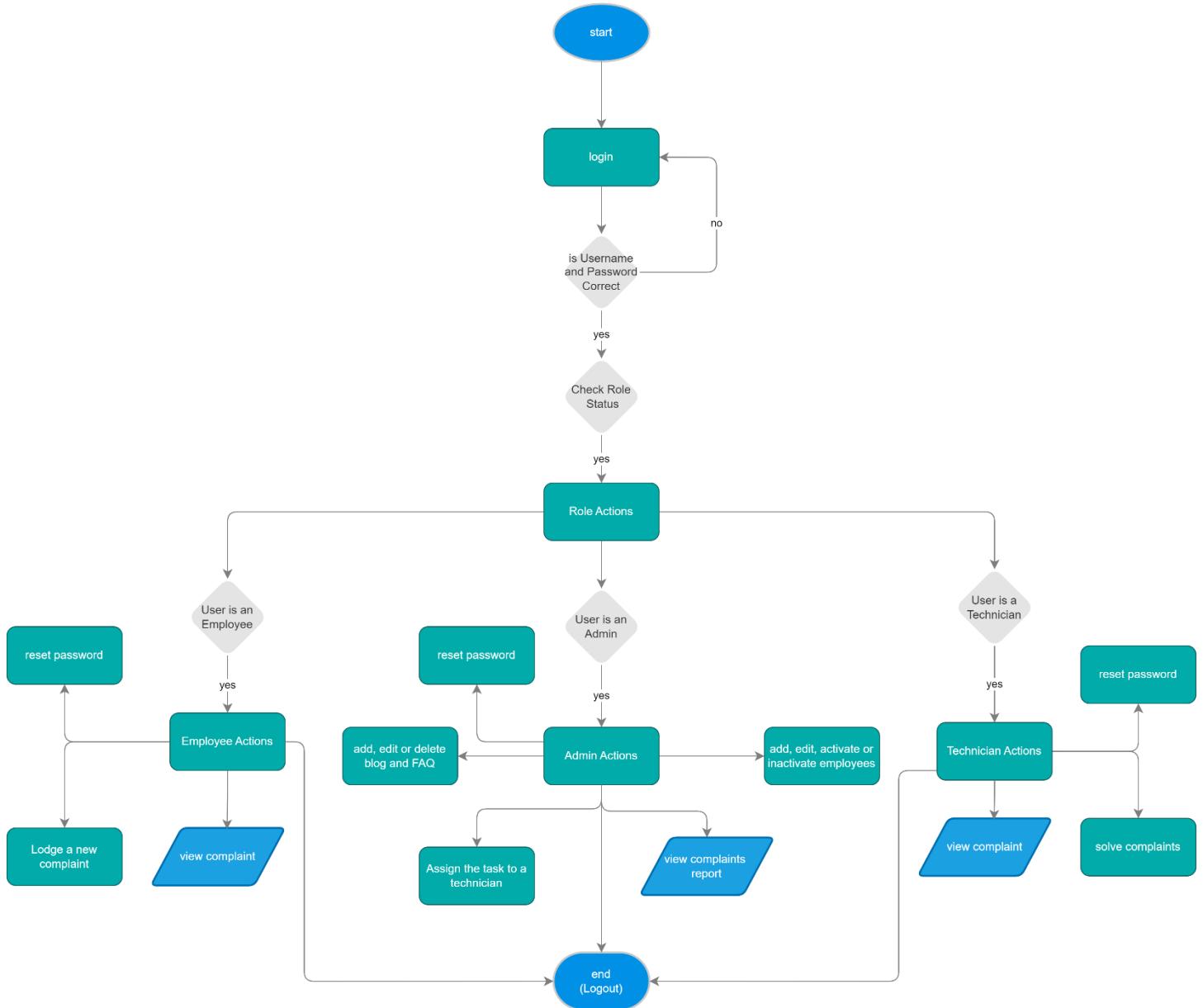
- Level 1 DFD:



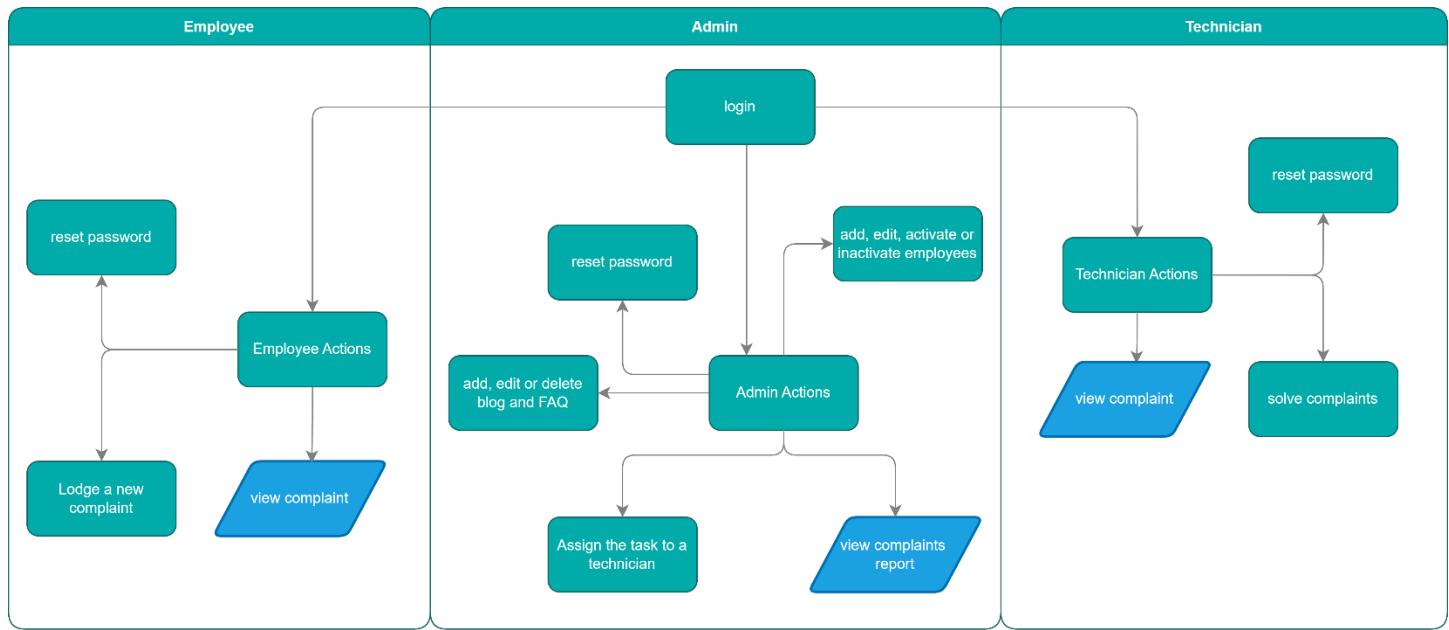
- Level 2 DFD:



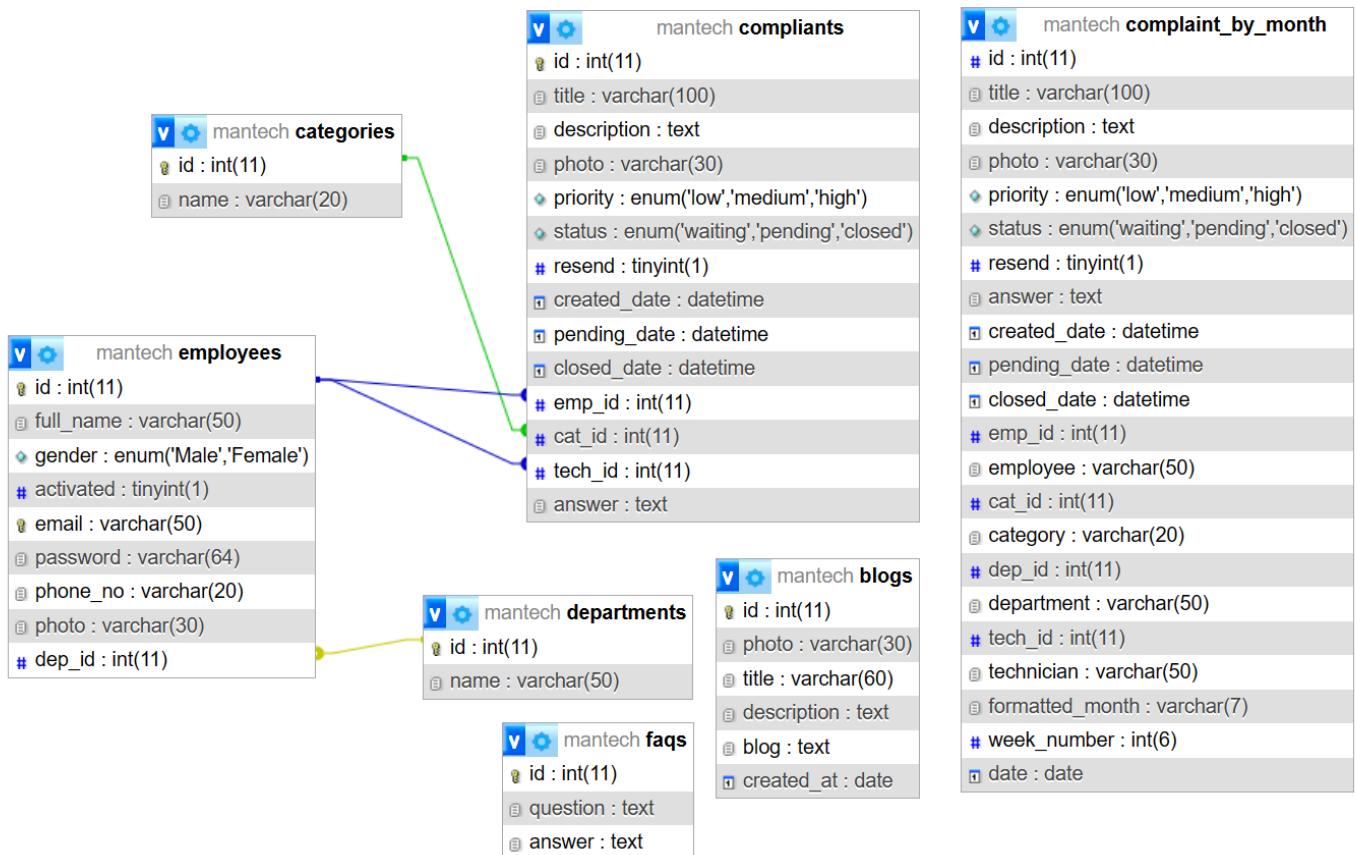
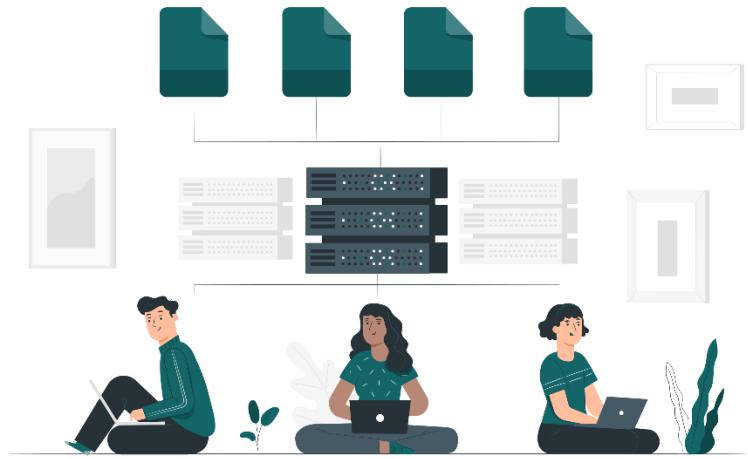
Flowcharts



Process diagram



Database design/ structure



Screen shots

Testing



-Reviewing comments and coding conventions:

No	Aspect Test	Remark
1	Have all the methods been properly integrated and they are completely functional?	Yes
2	Does each unit meet its objective and purpose?	Yes
3	Are all the validations happening properly?	Yes
4	Is the code working as per specification?	Yes
5	Does the application's functionality resolve the client's problem and satisfy their needs completely?	Yes
6	Have the hardware and software been correctly chosen?	Yes
7	Is the code being tested?	Yes
8	Is the code being ready?	Yes

- Reviewing Error Handling:

No	Check	Remark
1	Are errors properly handled each time the function returns?	Yes
2	Are error messages conveying what exactly has occurred?	Yes
3	Has error handling code been tested?	Yes

- Reviewing comments and coding conventions:

No	Check	Remark
1	Does the code respect the project coding convention?	Yes
2	Are the variable declarations properly commented?	Yes
3	Are all functions, method, class, and resource files documented?	Yes
4	Are functions, methods, and classes really doing what the documentation says?	Yes

-Reviewing Code Main Functionalities:

No	Check	Remark
1	Can admins view complaints reports?	Yes
2	Can admins assign complaints to a technician user to solve?	Yes
3	Can admins create new employee accounts?	Yes
4	Can admins receive all complaints from all employees and sort them?	Yes
5	Can employees create new complaints?	Yes
6	Can employees resend complaints that have been pending for more than 2 days?	Yes
7	Can employees view all their complaint history/logs and details for each complaint?	Yes

Source Code

NOTE: The source code is located in this folder [mantech](#)



User Guide

NOTE: The user guide is located in a separate pdf file named [User_guide.pdf](#)



Developer's Guide

The project has been divided into two parts:

1. Backend (java folder):

❖ Controller:

- which contains the java code that fetch data to all the xhtml view pages.
- It includes these files:
 - BlogsManagedBean.java:
 - which handle data for the blogs web pages for the admin user.
 - BlogsFAQManagedBean.java:
 - which handle data for the blogs and frequently asked questions view web pages for all users.
 - CategoryManagedBean.java:
 - which handle data for the category web page for the admin user.
 - ClosedComplaintManagedBean.java:
 - which handle data for the closed complaints web page for the admin user in the reports section.
 - ComplaintDetailsManagedBean.java:
 - which handle data for a specific complaint's details web page for the admin in the reports section.
 - ComplaintManagedBean.java:
 - which handle data for all complaint's details web page for the admin in the reports section.
 - EmailSender.java:
 - which handle data of the complaint that will be sent to the admin email.
 - EmployeeComplaintManagedBean.java:

- which handle data of the employee complaints in the employee web pages.
- EmployeeManagedBean.java:
 - which handle data of the employee web pages for the admin user.
- FAQManagedBean.java:
 - which handle data of the frequently asked questions web pages for the admin user.
- LoginManagedBean.java:
 - which handle data for the login web page for all users.
- PendingComplaintManagedBean.java:
 - which handle data for the pending complaints web page for the admin user in the reports section.
- ProfileManagedBean.java:
 - which handle data for profile web pages for all pages.
- SummaryComplaintManagedBean.java:
 - which handle data for the complaint's summary web page for the admin user in the reports section.
- TechnicianComplaintManagedBean.java:
 - which handle data of the technician complaints in the technician web pages.
- WaitingComplaintManagedBean.java:
 - which handle data for the waiting complaints web page for the admin user.

❖ Entities:

- which contains the java classes that represent the tables in the database to use them with the JPA.

- It includes these files:

- Blogs.java.

- Categories.java.
- ComplaintByMonth.java.
- Compliants.java.
- Departments.java.
- Employees.java.
- Faqs.java.

❖ Filters:

- which contains the java classes that contains filters to restrict users' access permissions.

- It includes these files:

- AdminFilter.java.
- EmployeeFilter.java.
- TechnicianFilter.java.

❖ Model:

- which contains java classes that handle database operations (insert, update, delete, select) done in the entities.

- It includes these files:

- AbstractFacade.java:
 - which contain all common database operations to be inherited in all other class.
- BlogsFacade.java:
 - which contain operations done in the blogs entity.
- CategoriesFacade.java:
 - which contain operations done in the categories entity.
- ComplaintByMonthFacade.java:

- which contain operations done in the ComplaintByMonth view in the database.
- CompliantsFacade.java:
 - which contain operations done in the compliants entity.
- DepartmentsFacade.java:
 - which contain operations done in the departments entity.
- EmployeesFacade.java:
 - which contain operations done in the employees entity.
- FaqsFacade.java:
 - which contain operations done in the faqs entity.

2. Frontend (web folder):

❖ Admin:

- contains all the web pages for the admin users.
- It includes these files:
 - blogs:
 - it contains all the xhtml files to add a new blog, update an existing blog or view all blogs.
 - category:
 - it contains the xhtml file to add a new category, update an existing category or view all categories.
 - complaint:
 - it contains all the xhtml files to view all waiting complaints and view details of a specific complaint.
 - employee:
 - it contains all the xhtml files to add a new employee, update an existing employee or view all employees.
 - faq:

- it contains all the XHTML files to add a new frequently asked question, update an existing frequently asked question or view all frequently asked questions.
 - info:
 - it contains the XHTML files to view the admin profile web page.
 - report:
 - it contains all XHTML files for viewing reports.
- ❖ **blogs_FAQ:**
- contains the web pages for the blogs and frequently asked questions sections in the landing page.
- ❖ **employee:**
- contains all the web pages for the employee users.
 - it includes XHTML files for adding new complaints (add.xhtml), viewing employee complaints (view.xhtml) and viewing details of any specific complaint (complaintDetails.xhtml).
- ❖ **profile:**
- contains web pages for the profile page section.
- ❖ **technician:**
- contains all the web pages for the technician users.
 - it includes all XHTML files for viewing all the complaints assigned to the technician (view.xhtml) and viewing the details of any specific complaint to add an answer for it (complaintDetails.xhtml).
- ❖ **css:**
- contains all the style sheets files of the website.
- ❖ **js:**
- contains all the javascript files of the website.
- ❖ **img:**
- contains all the images required in the website.
- ❖ **upload:**
- contains all the images uploaded by users to the website.

❖ **fonts:**

- contains font files required in the website.

❖ **footer.xhtml:**

- contains the code for the footer section which is embedded in all website pages.

❖ **header.xhtml:**

- contains the code for the header section which is embedded in most of the website pages.

❖ **login.xhtml:**

- contains the code for the login page which is used by all users to get into their specific pages in the website.

