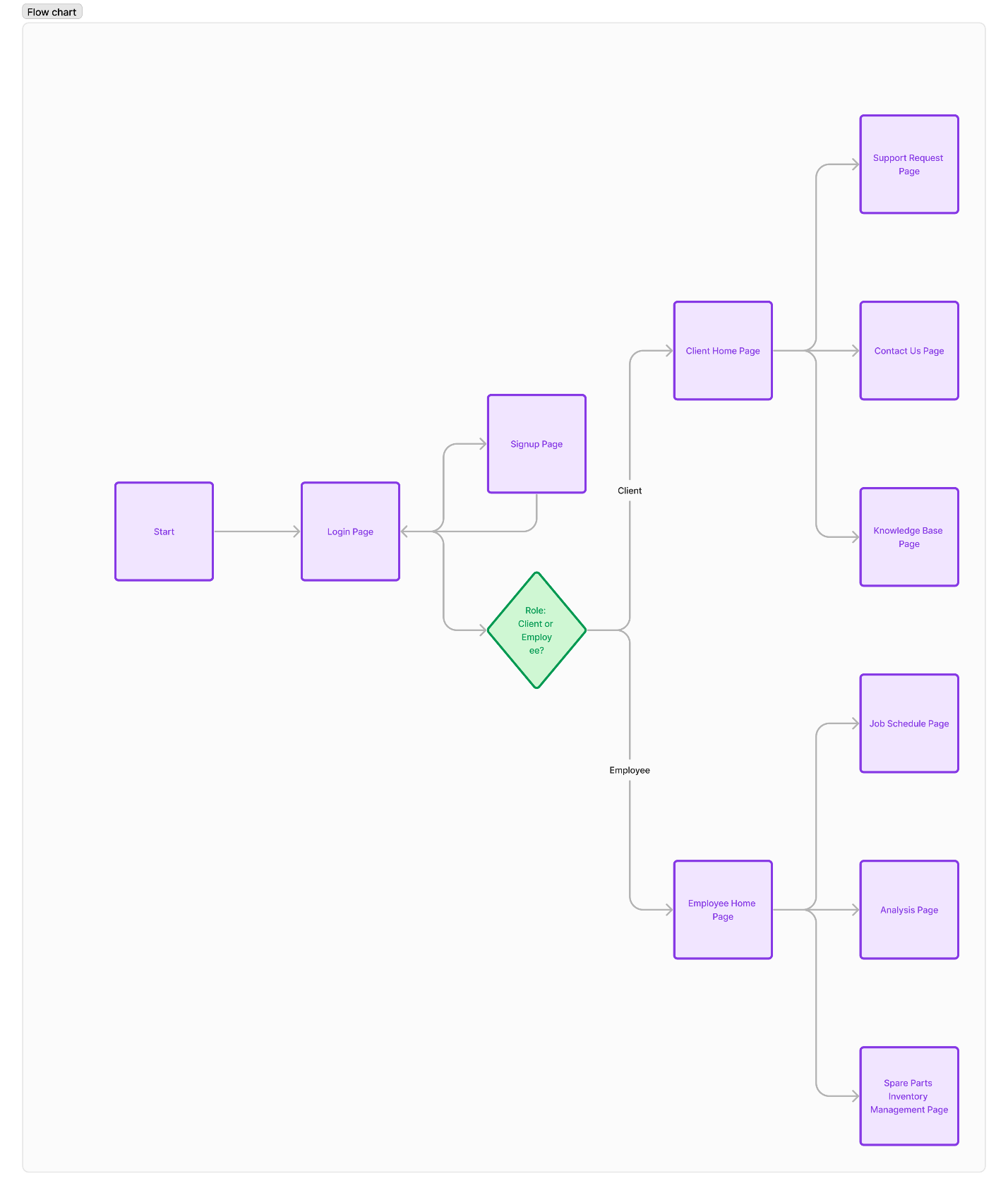
Business Context and Problem Summary:

* Dern-Support needs an IT Solution to cater to its growing needs, this IT solution is A web application, and its purpose is to help the company to accommodate to its ever-increasing demand.
* Developing a Web application will surely help the growing company to keep track of its customers’ needs and to help them grow their business even more.

Proposed Solution Description:

* The Full stack proposed solution is a website that will have multiple pages Those being:
  + A home page that differs based on whether the user is a new user, A returning user or an admin, every different page is different based on the role of the user.
  + A login page that sends you to the suitable home page based on the role of your account.
  + A sign-up page that always signs new accounts as Client accounts since Admin accounts are inserted into the database manually since its rare that you need to create a new Admin account.
  + If you enter as a client account then you will have the following pages available to you:
    - Support Request Page: this page allows you to submit your IT Support Requests and repairs.
    - Knowledge Base Page: this page gives you insights about some easy to solve problems that the user could solve on their own.
    - Contact Us Page: this page allows users to directly sends massages to the company, whether they are feedbacks or complaints.
  + If you enter as an Employee Account you will have the following pages available to you:
    - Spare parts Inventory Page: this page help you keep track of the spare parts inventory and see your stock of pc components.
    - Job scheduling page: this page allows you to add event to your schedule.
    - Analysis Page: this page allows the company to keep track of its growth over the years.

A flowchart that explains how the navigation works in the application:



* I Will use MERN stack to develop this full stack solution: MERN stands for MongoDB, Express.Js, React.js And Node.js
  + MongoDB will be used for the creation of NoSQL Databases
  + Express.js will be used for the creation of the Backend Servers.
  + React.js will be used for the creation of the Front-End
  + Node.js will be used for dependencies management and downloading.

Functional and Non-Functional Requirements:

* Functional Requirements:
  + A functioning Login page that redirects user correctly based on their roles in the database.
  + A Sign-up page that redirects users to the login page if their sign up was successful and inserts their data in the database and also warns them if their sign up was a failure.
  + A support request form that sends data to a database that will be used to help customers with their repairs
  + A table in the same page to the users what are their requests.
  + An inventory page for keeping track of the company’s spare parts
  + A job scheduling page for the ease of scheduling the jobs of the company.
  + An Analysis page to keep track of the company’s growth over the years and how it’s doing.
* Non-Functional Requirements:
  + A good and readable UI.
  + A good UX for the users.
  + A highly performing Website.
  + A good design.
  + Good security (Will be implemented in future plans)

Key Performance Indicators (KPIs):

* This will be based on an outside helper application like Google Analytics, so until the website is deployed, this will remain unchanged.

Risks and Implications:

* Project Delaying: might cause more unjust financial loses, because of the wages of the developers. To avoid such a trouble, we could allocate some of the budget in case a delay happens, so we could pay the developers.
* Possible PCs OS shenanigans: as the name suggests, PCs are not flawless, as they are always breaking and in need of repairing, if an important PC were to be broken, that would mean that some of the project data would be lost and possibly those are parts of the website so to avoid such a risk, it would be better to constantly backup your data across multiple PCs and on the cloud.
* Disapproval of the design: This suggests that the client might actually disapprove of the design, and to recover from such a risk, we could have a plan B design.