IT Management System

System Design Document

Latest Revision: 02/06/2022

PROJECT URL

https://github.com/comp195/senior-project-spring-2022-it-management-system.git

DEVELOPERS

Daniel Adler	(d_adler@u.pacific.edu)
Andrew Vu	(a_vu16@u.pacific.edu)
Vincent Tran	(v_tran37@u.pacific.edu)

System Architecture

This system is separated into two components: The software application used by high permission level internal parties in order to access and modify relevant information for the business, and the web application will provide other employees with the ability to submit tickets regarding maintenance and view relevant information.

Desktop Application

Software Modules

This software is composed of a client-side application where users can access and modify information about their businesses and their assets from both a management perspective and an employee perspective with different permissions assigned to each role.

Hardware Modules

This application will be designed to run on a computer running Windows 10 and above.

The User Interface Modules

- This application will consist of a series of panels that provide users with access to various different components within the IT management system.

- The **Login** panel will provide employees and management with the ability to log in with their own credentials, preventing people with insufficient permissions from accessing private data.

- The **Main Menu** panel is accessible to all users upon login and contains a variety of different tabs that they can click on in order to view and edit relevant information.

- The **Equipment** panel will provide employees with the ability to check equipment in and out while also allowing management to append or remove equipment from this stack.

- The **Tickets** panel allows the user to view all issue tickets submitted by staff and allows them to be sorted by priority, category, and user.

- The **Employees** panel provides information about each employee, including the equipment they've checked out, their IDs, timecard information, etc;

- The **IT Team Member Rating** provides managers with the ability to see how satisfied customers are with the resolutions to support tickets provided by the IT Team.

Interfaces to External Systems

The desktop client will need to access an external database in order to both retrieve and update information regarding tickets, employees, and other business information. The intended database for this use case is handled through Amazon Web Services (AWS).

Web Application

Software Modules

This part of the project contains the web client that is used by general employees of the company. It will be developed using HTML, CSS, and Python 3.0, and its main purpose is to provide employees with an interactive interface to communicate with management and/or the IT support team.

Hardware Modules

This web application will be designed under the assumption that the user is accessing it through a desktop browser (with a clear message notifying the user upon entering the application).

The User Interface Modules

- The **Email** text box serves as the main means of maintaining a unique identifier for the customer and will be a required field.
- The **Ticket Title** text box allows the customer to enter a short description of the issue to be addressed.
- The **Ticket Description** text box serves as the main point in which the customer will go further into detail and describe the issue that is to be addressed.
- There will be **Dropdown Lists** to help categorize and specify the customer's issue: **Request Type**, **Ticket Scope (individual, team, departmental, or larger)**, **Ticket Classification, etc.**

Interfaces to External Systems

The web client will need to access an external database in order to update information regarding customer tickets, which will then be retrieved and potentially updated through the desktop client's access to the data. The intended database for this use case is handled through Amazon Web Services (AWS).

Hardware, Software, and System Requirements

Desktop Application

Hardware Requirements:
RAM: 2 GB RAM
Storage: 1GB
Network: Broadband Internet connect

Software Requirements:
Python 3

System Requirements:
Windows 10/11

Web Application

Hardware Requirements:
Network: Broadband Internet connect

Software Requirements:
Web Browsers: Google Chrome, Firefox, Safari, Microsoft Edge, etc.
Python 3, HTML 5, CSS

System Requirements:
Windows 10/11, Linux, MacOS

External Interfaces

AWS:

AWS will host the database to store the information for our application.
https://docs.aws.amazon.com/rds/index.html
https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Welcome.html

Potential Web Hosting:

The web client may eventually expand and be hosted with an actual domain name, which would be handled through GoDaddy.
https://www.godaddy.com/

Fuzzy Search:

The library searches for matches and near-matches of a given text and allows us to create a search functionality within the application.
https://pypi.org/project/fuzzysearch/

Request:

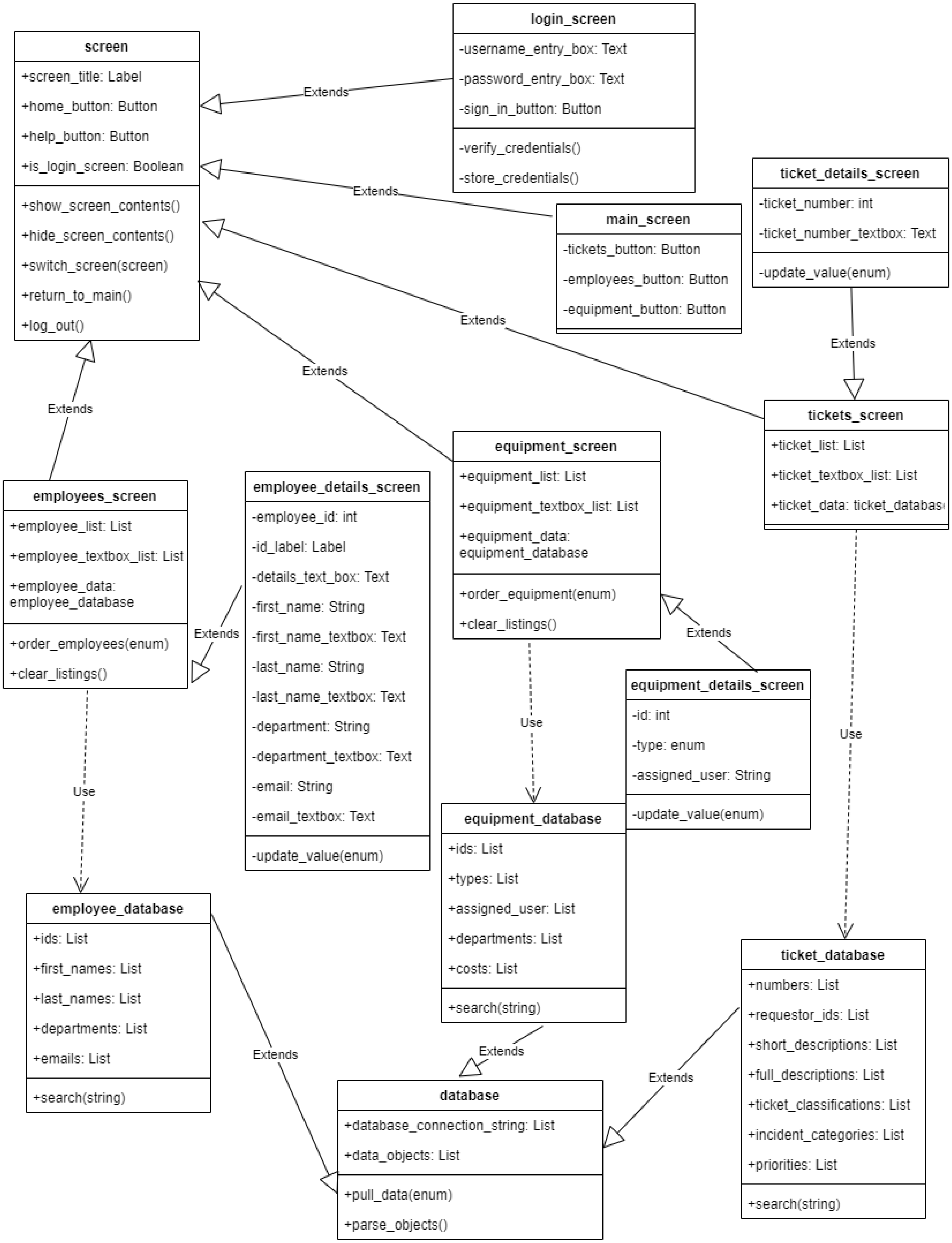
The **requests** library allows us to send HTTP requests which will be a part of the web application.
https://docs.python-requests.org/en/latest/

PyMySQL:

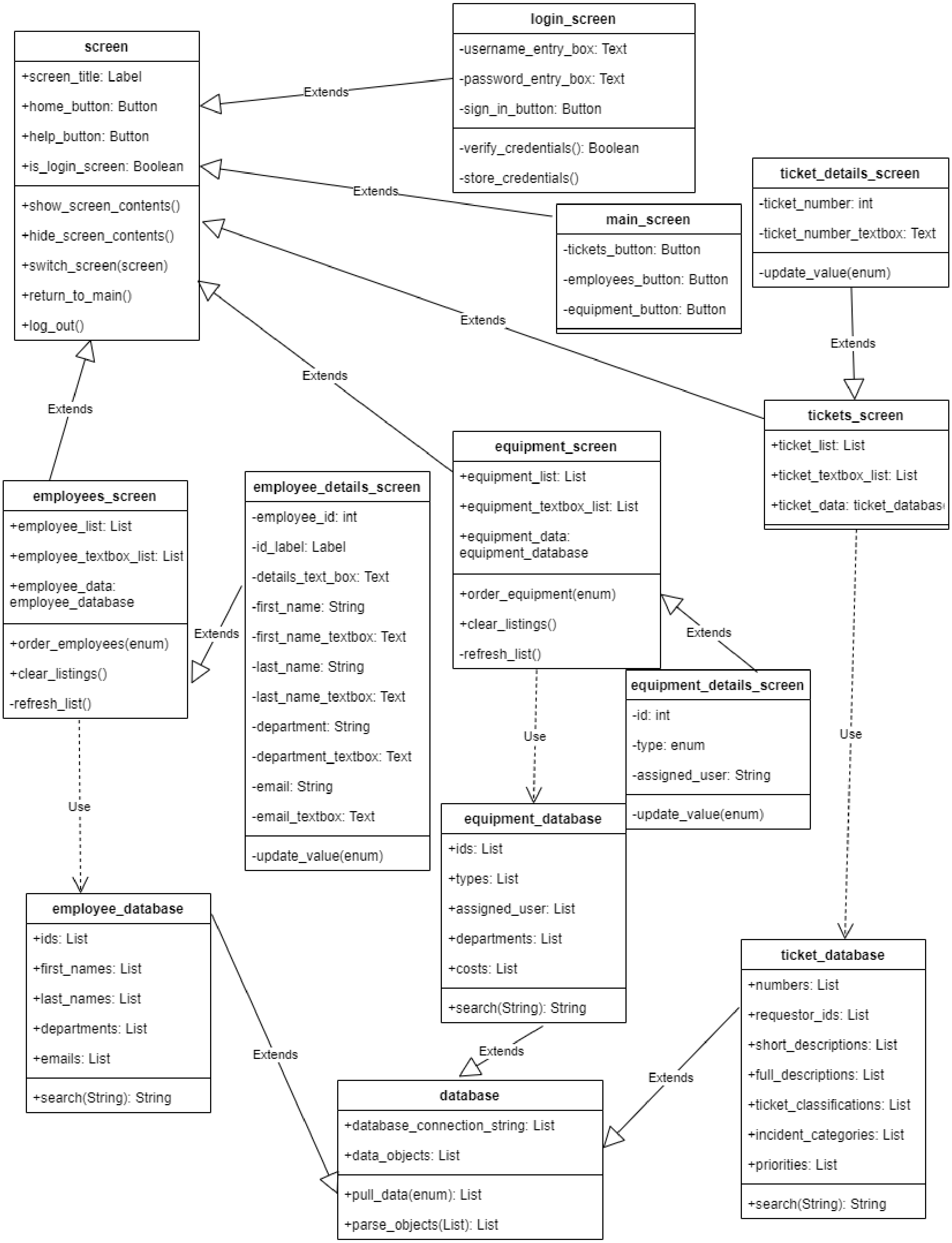
PyMySQL allows us to connect to a SQL database.
https://pypi.org/project/PyMySQL/

Software Design

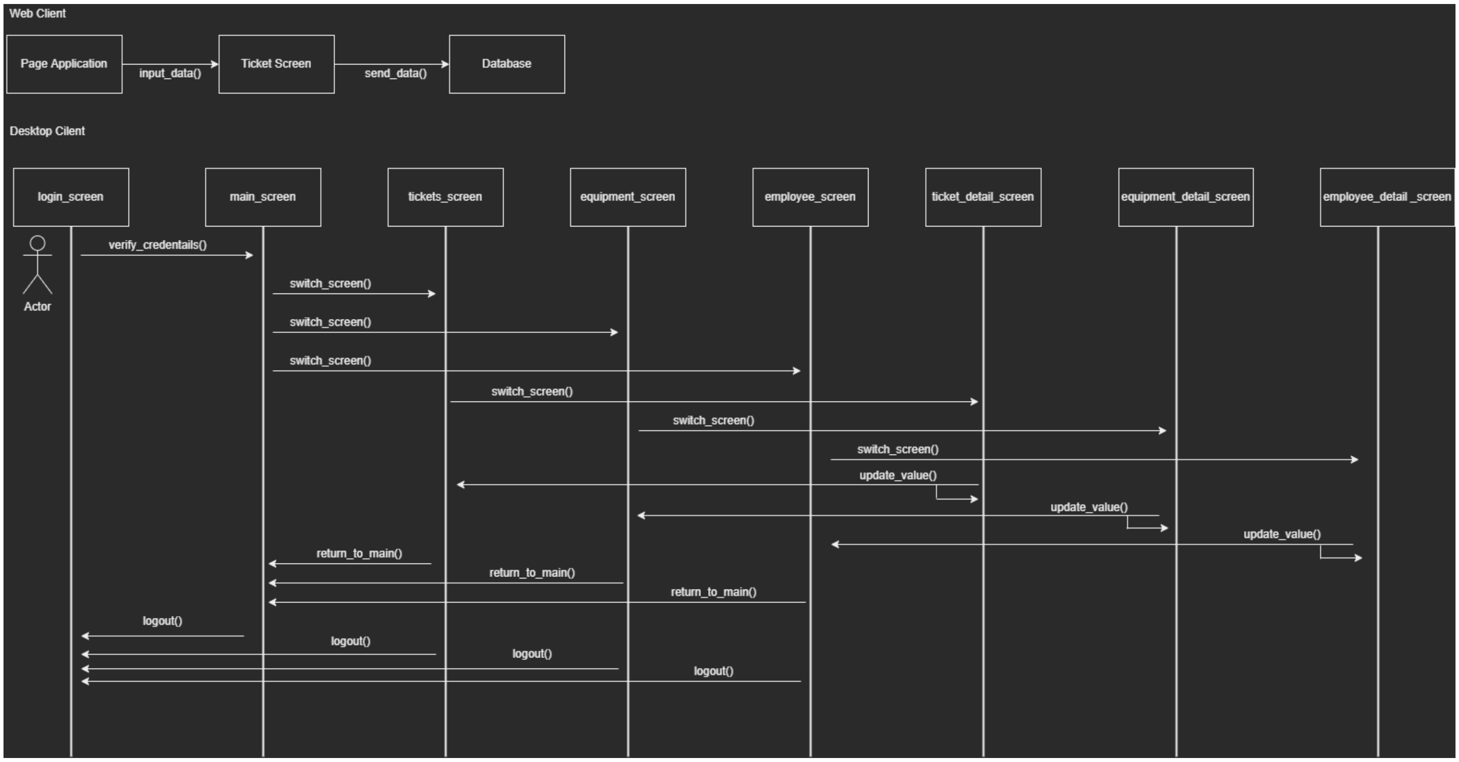
Class Diagram



Class Specifications



Interaction Diagrams



*Link to view the interaction diagrams in larger size
https://tinyurl.com/4hmkjb43

Design Considerations

Design Principle: Accommodate Change

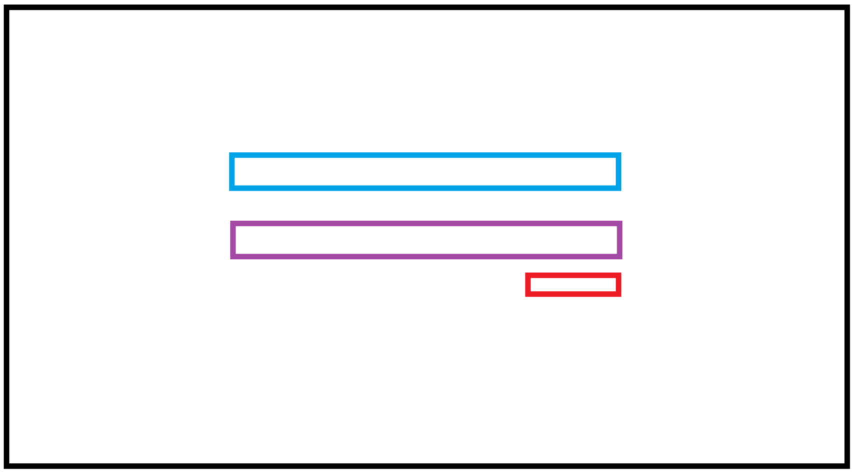
In designing the overall structure of the system, being able to accommodate for the change in the application's need of various business information details and components was taken into consideration. For example, while the IT ticketing system is a significant feature, a business's management may want to incorporate other management options to improve workflow.

Design Principle: Design for Testability

The idea behind the splitting of the system into three main components (database, desktop client, and web client) is that they can easily assist in the testing of each other's functionality. For instance, the web application's main purpose is to create a ticket, which can then be checked through the database for actual updates in the data. Furthermore, this can be confirmed through the desktop client's pulling of data from the database, and the desktop application's functionality of making changes to the central data can be checked back through the database as well.

User Interface Design

Login Screen



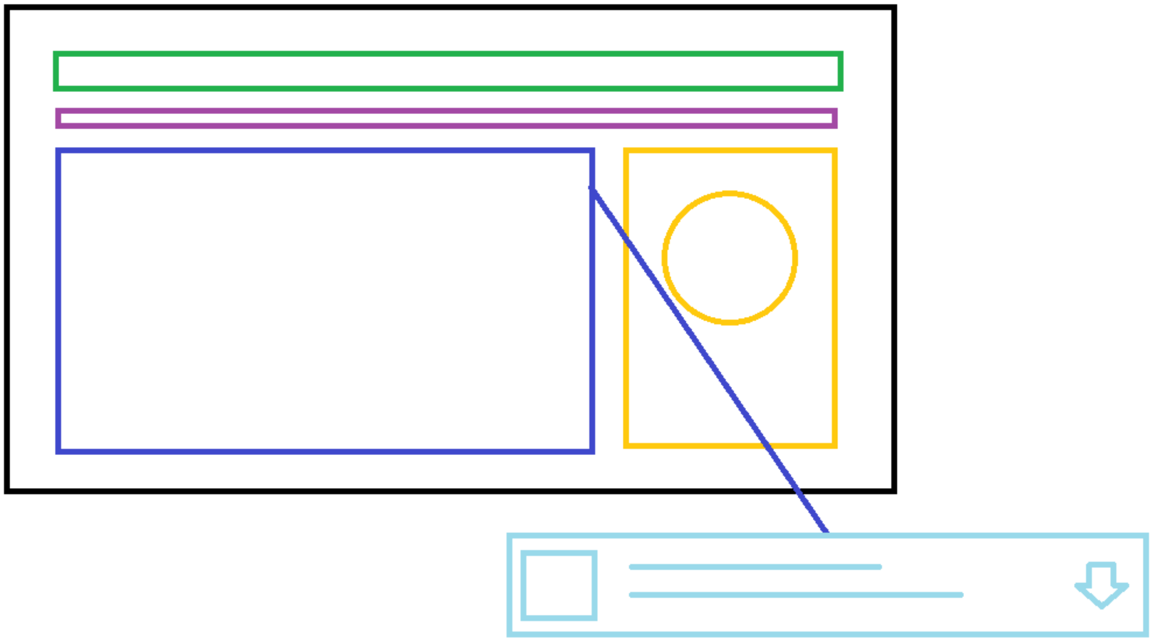
This screen will serve the purpose of allowing the user to login.

- **The Email Box**, indicated by the blue section, will allow the user to enter their email address.

- **The Password Box**, indicated by the purple section, will allow the user to enter their password.

- **The Forgot Password Box**, indicated by the red section, will prompt the user for their email address and send them a temporary password they can use to reset their password.

Database Screens (Equipment, Employees, IT Member Rating, Ticketing from IT Perspective)



The Database Screens will allow the user to see a variety of information about the assets and employees that a business has and make modifications to that data.

- **The Tab Bar**, indicated by the green section, will allow users to select which information they want to view or modify. These will be labeled as Equipment, Employees, Ticket, etc.

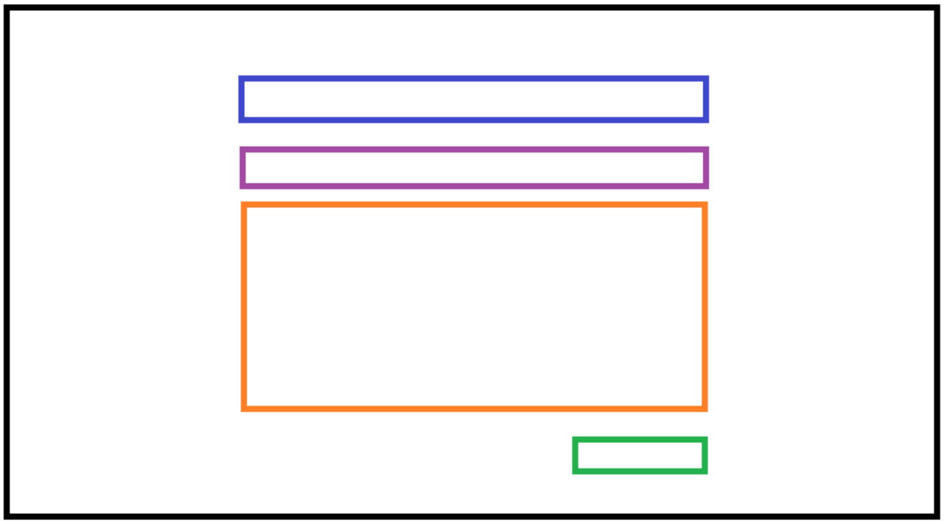
- **The Search Bar**, indicated by the purple section, will allow users to try to find an item within the list view based on a fuzzy searching algorithm by typing in either a keyword or item name.

- **The List Box**, indicated by the blue section, will show users a list of item boxes based on the contents of the search bar.

- **The Item View**, indicated by the cyan section, is found in multiple quantities within the List Box and displays minimal information about an item such as its name, image, and description. Clicking on the view will cause more information about that item to appear within the Details Box and clicking on the arrow will allow users to modify information associated with that item if they have sufficient permissions.

- **The Details Box**, indicated by the orange section, displays the currently selected item in the item view in more detail.

Ticket Screen



The Ticket Screen provides users with an interface to submit tickets regarding any issues they may be experiencing.

- **The Title Box**, indicated by the blue section, is a text box that will allow users to give their issues a title.

- **The Category Box**, indicated by the purple section, is a dropdown list that will allow users to categorize their issues for the purposes of filtering by the IT perspective.

- **The Description Box**, indicated by the orange section, is a text box that will allow users to describe the issue that they are having in more detail.

- **The Submission Box**, indicated by the green section, will consist of three buttons. The first button will allow users to upload an image that might help IT or management understand the issue in more detail, the second button will allow users to discard their ticket if they decide that they do not need to make the submission, and the third button will allow users to submit the ticket which will make it appear to users with sufficient permissions to view support tickets.

Glossary of Terms

Application (n.)
A program that runs on a computer. Software applications run on the desktop while web applications run on a browser.

Dropdown List (n.)
A user interface component that allows the user to select an option from a list that is revealed to the user upon clicking on a labeled button.

Incident (n.)
A ticket type in which the user is experiencing a problem (as opposed to a request). For example, an internet outage issue is classified as an incident.

IT Team (n.)
The individual or group of individuals responsible for the maintenance of data stored within this application.

Log in (v.)
Entering username and password credentials for the purpose of only allowing users with a sufficient permission level to access the data available to those at that level.

Management (n.)
High-level parties who are responsible for the assets and processes that occur within an organization.

Panel (n.)
A user interface component that represents everything visible to the user at a specific point in time.

Permission Level (n.)
A value that represents the amount of information that a specific user has access to. A higher permission level indicates that the user has access to more data and vice versa.

Perspective (n.)
Describes what each user sees within the software application. A user with a higher permission level will have a wider perspective than a user with a low permission level.

Request (n.)
A ticket type in which the customer needs an additional service; for instance, requesting upgraded equipment classifies as a request.

Ticket (n.)
A document submitted by a employee within the organization that references a specific issue said-user may be having with an asset or process within that organization.

Ticket Scope (n.)

The range of impact in which the issue has on a customer/user-base. For example, a scope value of “departmental” indicates that the issue affects the department as a whole and is therefore more serious than a ticket labeled with a scope of “individual”.

User (n.)
Any individual that utilizes the functionality offered by this application.

References

Okyol, O. (2020, August 4). *Using AWS RDS and python together*. Towards Data Science. Retrieved February 7, 2022, from https://towardsdatascience.com/using-aws-rds-and-python-together-5718a6878e4c.

Principles of Software Design. GeeksforGeeks. (2020, June 17). Retrieved February 7, 2022, from https://www.geeksforgeeks.org/principles-of-software-design/

Python tkinter tutorial. GeeksforGeeks. (2021, March 28). Retrieved February 7, 2022, from https://www.geeksforgeeks.org/python-tkinter-tutorial/.