A picture containing text, tableware

Description automatically generated

**פרוייקט מעשי לתואר B.Sc. במדעי המחשב**

**קמפוס חכם**

**Smart Campus**



**מגישים:**

מורן רון אנרי בן ציון ליאור אברהמי

לינוי רון אור אבוטבול שיר דוידוב

**בהנחיית:**

שירן יוסף

**Table of contact**

**Cover page…………………………………………………………………. 1**

**Table of contact ……………………………………………………. 2**

**Project characterization …………………………………… 3-4**

**System requirements…………………………………………….5-9**

**Project characterization**

**Project name**

smartCampus

**Project description**

A real-time system with an interactive web interface that monitors the various activities on campus, and allow the user to get information about the different facilities on campus.

The system includes management and control of various communication components within the organization and in real time.

**Applicability research**

* The system is designed to enhance the administrative management experience in the organization.
* For every smart component in the organization, the system will contain a service that will enable the transfer of data and communications in front of the same component. In addition, the system will interface with a database which will mainly hold the relevant data that were sent from the smart components.
* The system will enable the production of statistics on the organization's use of the various components. which

will allow for in-depth learning about the consumption habits of the organization and ultimately for efficiency

and cost reduction.

* In addition, the system will also provide information to campus users, students and lecturers about available parking spaces, list of chillers, the number of attendees in a particular building and classroom and will provide help with on-campus navigation.

**Stakeholders**

* Direct stakeholders

Administrative manager - the system will make it possible to optimize the organization's consumption.

As a result, it will reduce administrative management times. shortcut the length of time involved in managing the organization will allow for a focus on other issues of interest.

* Indirect stakeholders
  + **Residents of the organization -** shortening the

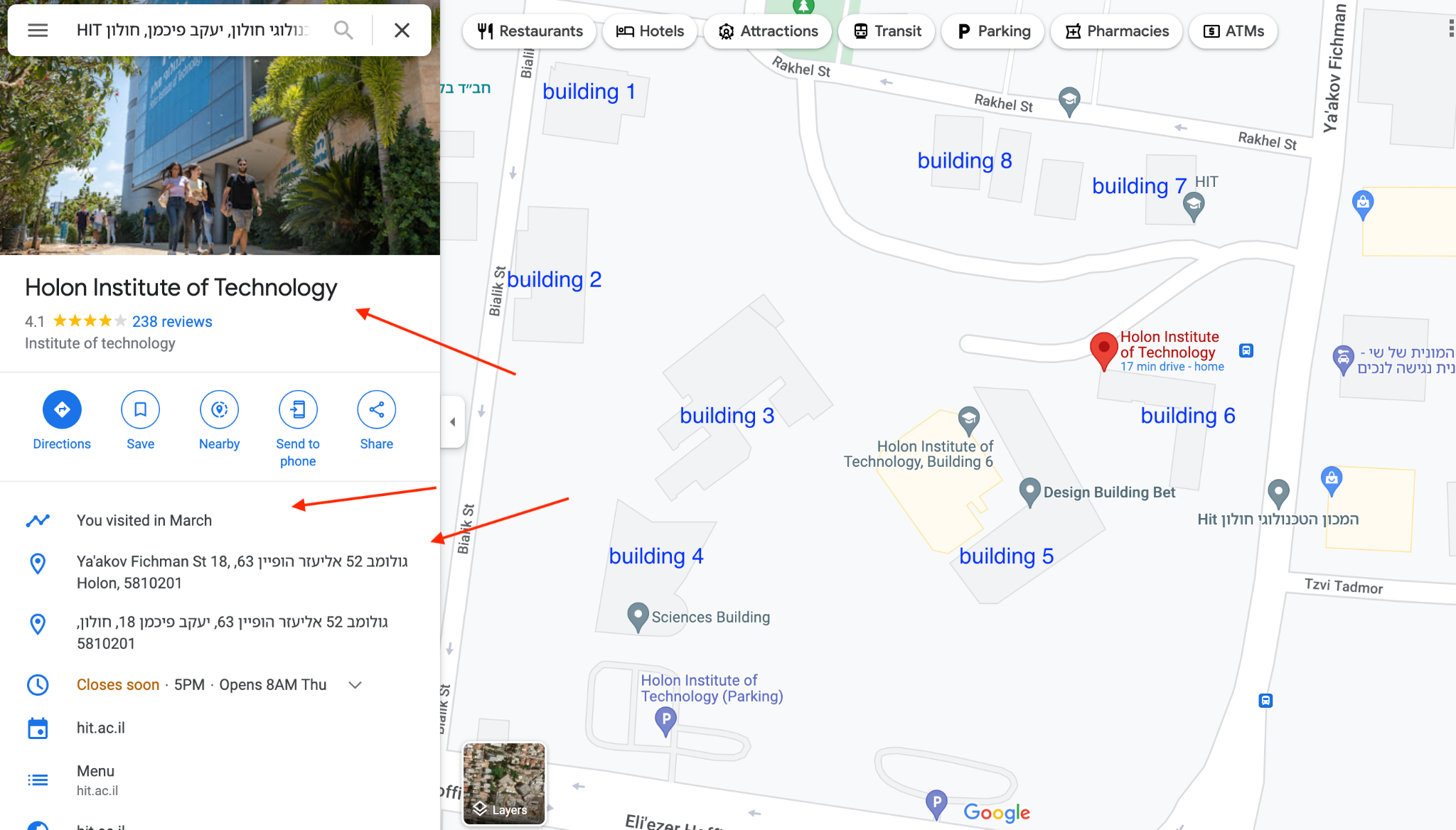
management times of the organization. It will lead to improved organization conditions and satisfaction

desire on the part of the tenants present in the organization.

* + **Organizational Management Level -** Accurate statistics on organizational habits (use electricity, water, etc..) will reflect the real needs and even provide an overall picture of these needs and will lead to the efficiency of the organization.

**System requirements:**

1. **Functional requirements:**

* The system will display on the user interface how much electrical charge is used in a given classroom/building in real-time, this information will be available for the management.
* There will be a module which will open a map of the HIT campus (like google maps). Each click on a site on the map, will lead us to a data page about the same building(such as electric usage, and water usage) for example: 
* If the user wants, The system prints a daily report on the usage of electric, and water usage (and is shown to the admin) with a push of a button. (this feature is relevant for the management).
* There will be a sub-module which enables the management to add, delete and change chillers on the Data-base.
* There will be an accessible tab that will give you input about available parking spaces (will be applicable in the future).
* There will be a tab informing about the number of attendees in a classroom/building (will be applicable in the future).

1. **Functional requirements (things the system cannot do):**

* The system cannot provide any change to the data(chillers, water running, electricity usage) in the DB via the user- interface.

1. **Access Control:**

|  |  |  |  |
| --- | --- | --- | --- |
| screen / action | IT (Admin) | Lecturers | Student |
| User Management - Add | V |  |  |
| User Management - Edit | V |  |  |
| User Management - Delete | V |  |  |
| User Management - View | V | V |  |
| Location Management- Add | V |  |  |
| Location Management - Edit | V |  |  |
| Location Management - Delete | V |  |  |
| Location Management - View | V | V |  |
| Chiller Management - Add | V |  |  |
| Chiller Management - Edit | V |  |  |
| Chiller Management - Delete | V |  |  |
| Chiller Management - View | V | V |  |
| Counter Management- Add | V |  |  |
| Counter Management- Edit | V |  |  |
| Counter Management- Delete | V |  |  |
| Counter Management- View | V | V |  |
| Reports | V | V |  |
| Statistics | V | V |  |
| Predictions | V | V |  |
| Alert management | V | V |  |
| Available parking spots | V | V | V |
| Available class in each building | V | V | V |
| Access to water meters | V |  |  |
| Air pollution monitor | V | V | V |
| Access to electricity meters | V |  |  |
| Sprinkler control | V |  |  |
| Counter of people in each building | V | V | V |

**3. Non-Functional requirements:**

* The system will classify users into students, lecturers and administration.
* The system will grant admin privileges for management.
* In case of connection loss, the system will use the latest data and show an appropriate message notifying about the connection lost.
* Pop up password verification when managing components on each group of sensors.
* There will be a different password for the DB itself , and it will be granted only to the system admin/s.
* Screen refresh time will be every 5 minutes.
* The system availability is 24/7.
* The system will be under maintenance on weekends, urgent cases will be taken care of immediately.
* The system should store 2 copies: 1 for production (functioning system) , and 1 for Dev & Test.
* The system will be able to handle multiple requests and multiple users at the same time.
* The database data should be updated every 10 minutes.
* The database holds the data collected from the chillers and all sensors for 3 years, and will delete all data each 3 year cycle.
* The system language will be English (Hebrew can be added later).

**4. Tech stack:**

* Front End - Javascript & React
* Back End – Python
* Database - POSTGRESQL
* Cross platform- web, mobile (iOS, Android)