Self Service/Monitoring Garage Portal
AKA: Garage-K9

Ahmed Hassan
*Deparment of Computer Science and
Information Technlogy
Univeristy of the District of Columbia
Washington,DC. , United States of America*
ahmedsabry.hassan@udc.edu

Abstract— **Developing a web application requires client server architecture, full stack development which consists of four major components, from front-end, back-end, databases to a server. In this paper, I discuss the functions which I have implemented as well as future functions, and how they will become an asset in simplifying purchasing a parking pass while providing the ability to distinguish between a student's car and those who are coming from the outside to use this public parking.**

Keywords—HTML5, CSS, JavaScript, Static Webpage, Dynamic Webpage, HTTP, Web Authentication, Hosting

I. INTRODUCTION

The World Wide Web is the most used service on the internet, allowing people to search, obtain and share data. Therefore, it is the most influential platform used nowadays. The web is a virtual system that interconnects resources on the internet such as documents, images, videos, and many other resources via hyperlinks, referenced with URIs (Uniform Resource Identifier). [1]

Web browsers such as Chrome, Internet Explorer, Firefox, Safari, and many other similar programs are used to access web resources. Usually, these resources are presented in web pages and sometimes in structured interrelated group of web pages called websites. Web pages are created using HyperText Markup Language (HTML) for structuring the page, CSS to style it, as well as using scripting languages such as JavaScript to add interactivity to the web page and make it more user friendly.

II. BACKGROUND

A. Purpose of my web application

As web applications continues to advance, they have become more accessible than ever, it is allowing us to solve complex issues and simplify tasks. Witnessing and experiencing the process of purchasing a parking pass at UDC seemed complex and outdated as it requires a few steps as well as visiting multiple buildings. After looking back at these steps, I decided to develop a web application to assist in simplifying this process as it is one of many purposes and goals of this tool.

Rather than having to visit the cashier's office, fill out handwritten forms, make in person payment. Instead, students shall purchase parking passes online using a portal and be billed through their student account or check out using a credit card. This can be done from the comfort of their mobile devices. Once, students register on the portal by creating an account, purchase their parking pass, which requires students to add their vehicle information such as the vehicle's make and model as well as their license plate number. This information is then linked to their account on the database. After parking pass is successfully purchased, a QR code would be generated, and sent out as part of a confirmation email. This QR code will be sent out to student's email which contains a PDF formatted file that acts as a confirmation. Later, students can print out their QR codes as well as placing them on their vehicle's windshield. This QR code shall be designated to them as it will be associated and linked to their N Number throughout their time at UDC.

Aside from simplifying the process of purchasing parking passes, this web application acts as an inventory management system as it can used to track and keep record of student parking passes distributed each semester at UDC parking garage. This can provide the school with data such as the number of passes distributed each semester, and track which passes are assigned to whom, and to which vehicles. This could be an asset to campus police and garage staff when it gets to distinguishing between a student's car, and those who are coming from the outside to use this public parking. This could also help in solving the issues of unknown vehicles that are parked overnight or in spaces which they are not allowed to park in.

B. Main ideas and scope

Portal: in this portal tabs such as "Purchase Parking Pass", "Report Incident" and "Contact" can be found and accessed through the homepage as shown in figure.1 below. Each tab/hyperlink is linked to a form which directs students to the appropriate task.
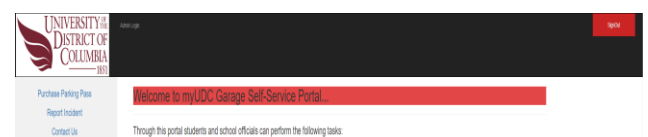

Figure 1. This is the homepage of the portal which students would see as they visit.

Student Login Page: Once students have sucessfuly created an account then they will be able to login using their email and password. As they sucessfuly log in, then they would be able to view their profile as well as having the ability to edit some information while some information would stay fixed and restricted. Some of the information students can possibly edit their own could be their vehcie's information and have the option to "add" or "delete" vehicle. This can be useful in cases where students may drive more than one vehicle as both vehicles information will be registered in the database regardless.

Figure2. Student login page and actions they can take once they are logged in.

Report Incident function: This page will allow students to report incidents they may have witnessed anonymously without the need of having to login or include any information that would identify them. This will be done through filling out a form indicating the incident date as well as typing information about it. There is also an attach pictuers field which would allow students to upload pictures they may have captured during the incident which they witnessed.

Figure3. Report incident form which can be used to annousmly report witnessed incidents

Purchase Parking Pass form: This form can be accessed on the homepage once an account has been created. Students will be asked to fill out contact information as well as providing information about their vehicle. This will be done by typing the input required field as well as choosing options from select menus and sub menu. They will be asked which semester they wish to purchase a parking pass for, whether it's "Fall" or "Spring" or "Academic Year" They will also be asked if they live on campus and require an overnight parking pass. Upon submitting this form, students will have the option to either be billed through their student account or have the option to check out using PayPal or a a credit card as show in figure below.

Figure4. Parking Pass Purchase/Pay Now feature if students wish to checkout immediatley using Paypal or a credit card.

Inventory management system: Since this web application is based on forms which are linked to a database. The school can always refer to the database and gather the data needed, as necessary. Some of these data assets which can pulled from the database are, the number of passes distributed each semester, number of students and vehicles assigned to them.

C. Admin portal

There are three main functions which can be used from the admin portal by those who have an admin access privilege.

Look Up Vehicle/Checker function: This allows campus police and garage staff to quickly identify the vehicle's owner in case of emergencies. This is done either by scanning the QR code placed on the vehicle's windshield or by manually typing the license plate number and click on "Search Vehicle". Once done so, an output will provide the vehicle's information such as make, year and model as well as its VIN and license plate number. It will also display information such as to whether the student has a designated parking spots, and whether they are permitted to park overnight. It also displays the student's contact information which can be used to "Call" or "Alert" if the application is accessed through a mobile device, then by clicking on "Call" the user will be

prompted to making a call. While clicking on "Alert" would send a prepopulated email to one or many students using an email list. This is this useful in a scenario where construction work is scheduled to take place in certain section of the garage such as "B1" for instance. Then, admins would have the ability to send an alert ahead of time requesting the vehicle owners to relocate their vehicles elsewhere until further notice.

Check Vehicle

Enter Licence Plate Number [YF5590]   State [DC ∨]   [Check]

Figure5. Check/ Look Up vehicle function.

Fine function: This allows campus police or those who are with admin privileges to fine a vehicle using form which includes selections from the menu such as fine amount and the reason for it.

Fine Vehicle

Enter Licence Plate Number [YF9950]   State [DC ∨]   Amount [$25 ∨]   Reason [Parking in a handicap spot ∨]
[Fine]

Figure6. Fine function which uses input field and drop down selection menu

Send warring/Alerts: This function can used to send out either pre-populated or custom emails to students using an email list. This can be used to warn students of a violation they committed as well as sending out alerts to students in case of emergencies.

Search, Add, Update, Delete: This function allows admins to manually add a student record which would contain their contact information as well as their vehicle information. This is added using students N number. Student information is fixed while the vehicle's information is updatable as needed.

Fig7. Admins can manually add, search, update, delete student record using students N Number.
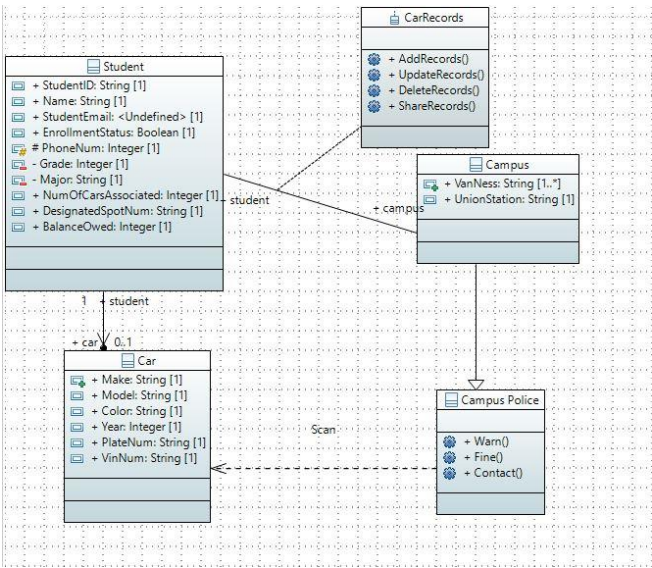
Figure8. The abov UML diagram showing the structures of my web application and the relationship between each class.

III. RELATED WORK

A. HTML5

With the fast advancement of technology, the demand for a better coding experience in creating webpages has pushed HTML to its limit and thus a better version was created in the new standard of HTML5. This standard support new elements, new attributes, local SQL databases and web applications.[2] Some of the features added to make the development process easier are local data storage which allows web pages to store information locally within the client browser. It also supports local file access as well as the use of local SQL databases and application caching which allows web applications to be accessible without an internet connection. There are also new form elements such as new input types and automatic validation.

Values of the target Attribute [3]

Value name	Function
_blank	Opens the linked document in a new tab or window.
_parent	Opens the link in the parent frame. Frames are deprecated in HTML5
_self	Open the link in the current frame.
_top	Opens the link in the top-most frame. Frames are deprecated in HTML5

Attributes of the anchor Element

Attribute name	Values	Function
hreflang		Specifies the language of the linked resource.
download		Directs the browser to download the linked resource rather than opening it
target	_blank _parent _self _top _frame name	specifies the context in which the linked resource will open
title	text	Defines the title of a link, which appears to the user as tooltip.
href	url	Specifies the linked document, resource, or location.

Static VS. Dynamic websites:

Static websites are websites that contains only static web pages which is a document that, when it is requested, displays on the browser exactly as it is stored in the server. Its content does not change unless the creator changes it manually. The URL address must end with a document extension such as .html or .php. However, if it does not then it is not considered a static page. [4]

1. Every page will have the code written separately as seen on the web.

2. Every page must be saved separately on the server.

3. Changes must be made manually every time, and coding knowledge is needed to make these changes.

Figure9. A diagram showing the process of getting a static web page displayed in the browser goes.

Dynamic web websites are websites that contains dynamic web pages, namely they can be generated using scripting languages and programs interacting with the database on the server side. They are programs that can add, restrict, edit, or remove data without the creator intervention. [5] This means that these webpages may change based on the user action or his computer information. For example, these pages may contain the current data and time at which the user is viewing the webpage, or the website can display a user's profile information when they log into it.

Figure10. A diagram showing which illustrates request in a dynamic webpage.

Static vs Dynamic

Static	Dynamic
HTML + CSS + JavaScript	Server-side language such as Node
Not dynamically rendered on a server, but still served by a server	Returns dynamically generated HTML page
Page content can change via JavaScript	Is not involved in page rendering after serving it
Rendering happens in the browser => Higher reactivity but data needs to be fetched after initial rendering	Rendering happens on Server=> Finished page is served but needs to be generated first
Security can be more complex	Security tends to be easier
Static host suffices, no complex server-side setup required	Requires host which supports the chosen server-side language

Deployment

Static	Dynamic
Only a static host is needed (it only needs to server HTML, JSS, CSS)	Host needs to support the chosen server-side language (and version)
Static hosts tend to be cheaper; setup is typically easier	Dynamic hosts tend to be more complex to setup and can be more expensive
Examples: AWS S3, Firebase Hosting	Examples: AWS EC2/ Elastic Beanstalk, Heroku

B. Web Applications:

Web application is a software program which exists on the server and runs using a web browser, through a web page. It is created using a combination of programming languages and web application frameworks. It may use the ram; allows user interactivity and it is designed for many uses and used in applications such as Gmail, Facebook, and YouTube.
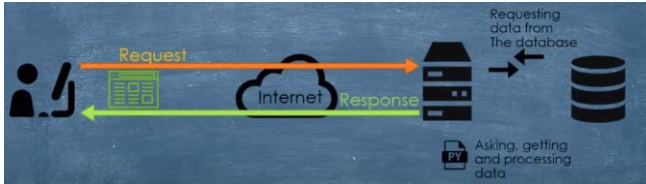

Figure11. This diagram explains how the process of accessing a web application which is similar to dynamic web pages request.

C. HTTP protocol

HTTP stands for Hypertext Transfer Protocol. It is an application layer protocol that allows web-based application to communicate and exchange data. It is a TCP/IP based protocol which Is used to deliver contents such as images, videos, audio, and documents.[6] HTTP was mainly designed to fetch html documents and sends it to the client. Since it was designed in a exquisite way, it has continually evlolved and features were being added to it. It has became the most convient way to quickly reliably move data on the web.

- HTTP is connectionless: after making the request, the client disconnect from the server, then when the response is ready the server reestablishes the connection again and deliver the response then it closes.

- HTTP can deliver any sort of data, if the two computers are able to read it.

- HTTP is a stateless as each request is executed independently, without the knowledge of the requests that were executed before it.

HTTP Messages

It consists of the three main sections such as start line, headers, and body. They all contain plain texted based information unless the body contains binary data. However, in general HTTP messages are plain text and easy to read. The information in the three section varies depending on the http message, whether it is a request or a response. A request http message differs from a response one and it contains the information such as follows in the figure below.
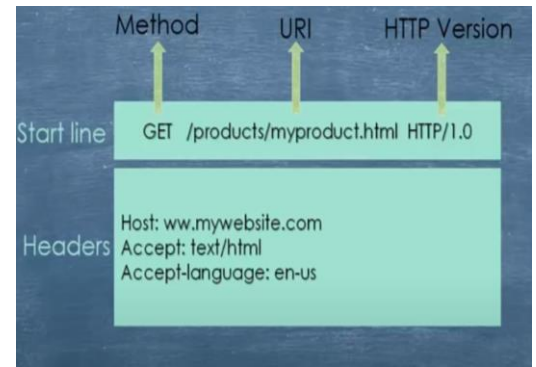

Figure12. Request in HTTP message

The method is a command that communicates with the server on how it should proceed. For example, this could be command such as provide data, delete a task, or add a task in the database. There are two well known HTTP methods such as GET which communicates with the sever to send data while the POST tells it to store data in the database.

D. Authnication with HTTP

HTTP basic authentication is the process in which the browser requests a username and password, in order to allow access to a specific resource. [7] For example, if a user is browsing and tries to make a request to the web server, to an URL called "home" since user is not authenticated then the browser responds with a 401-status code which means unauthorized. Then, by going to www-authenticate header. This instructs the browser to provide a pop-up in which we can write the username and the password needed to access the "home" resource. Once the username and password are typed, those values get concatenated and then encoded using base64 and then the browser sends the same request to home. However, this time the HTTP request will have a new authorization header which will have the value basic and then the encoded username and password. Finally, if those credentials were correct then the web server will validate them and then will send a response with the resource and a 200-status code which means Ok.
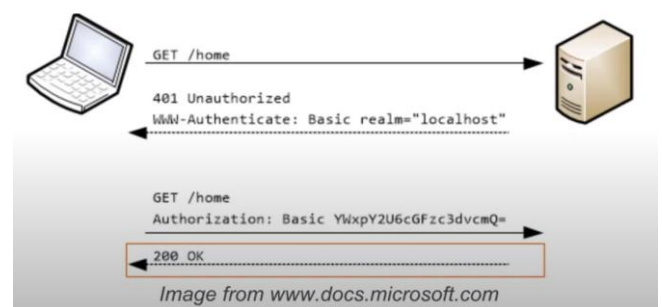

Figure13. The above diagram illustrates the process of authentication.

The process of Encoding of username and password

Step 1	Providing username and password ex. student/udc$21
Step 2	Username and password become combined as follows student:udc$21
Step 3	Encoding base64 which would look as follows: ZGFuOnBhc3N3b3Jk
Step 4	Finally, a new header will be added to the HTTP request which is an authorization, and its value looks as follows: Authorization:Basic ZGFuOnBhc3N3b3Jk

HTTP is the simplest form for authentication that can be used inside application. However, its transmitted credentials are not encrypted and only encoded with Base64 in transit and not hashed. This could be vulnerable as it can be easily decoded which would result attackers in finding out the username and password. It is recommended to use basic authentication in connection with SSL and HTTPS which would provide some level of confidentiality and further secure the basic authentication process.

E. Form-based Authentication

It is the process of authenticating a user by presenting a custom HTML page for user to enter their credentials.[8] Unlike HTTP basic authentication where a pop-up is displayed by the browser which is handled by the HTTP protocol. Next, those credentials are posted to the application as form data. The application is responsible for collecting form data and performing the authentication. This authentication works by using sessions and cookies.

F. Web Hosting

At the early stage of my web application development, my application was only a static website which means that it's files and content was only stored on my local machine rather than on the internet. This makes it limited as others would not be able to access and use my application online.

On this final stage of my development, I hosted my web application through Firebase which makes my web application accessible using a public URL< https://udcgarage-k9.web.app. This has made my web application to in becoming live which makes it easily accessible via mobile devices.

CONCLUSION

Throughout the development of this web application, I learned about the foundation which web applications are built on such as HTML, CSS, and JavaScript. I learned about Firebase which is Google's All-In-One backend solution as it provided me with all the features I needed such as Realtime database, authentication, and hosting to get my back end up and running. I also utilized some tools and learned how to use new ones such as GitHub (Version Control System). This allowed me to create repository, which contains all my project files and allows me to access each file's revision history as well as keeping a record of what changes were made. I can then revert to a previous version as needed.

Throughout the development stages, I used the best practices according to web standards such as authentication and security rules. As I continue to research and learn more about the upcoming features and enhancements that I would still like to implement. This tool will become UDC's All-In-One automated solutions for both staff and students as it will not only serve the purpose of simplifying the process of purchasing parking passes as well as obtaining overnight parking permits. In addition, it will also be an asset to security, allowing them to manage parking spaces at the garage in a more effective digitalized way. It will also help in solving the issues of unknown cars that are parked overnight or in spaces which they are not allowed to park in. It will limit students from violating parking rules as they would not be allowed to renew their parking passes if they are due for any fines. Overall, it will allow UDC to take a paperless approach in handling any garage related tasks.

References

[1]   S. J. Vaughan-Nichols, "Technology News," *Will HTML 5 Restandardize the Web?,* p. 14, 2010.

[2]   J. D. Eisenberg, "A LIST APART," 01 09 2009. [Online]. Available: https://alistapart.com/article/get-ready-for-html-5/.

[3]   A. Woods, "HTML," [Online]. Available: https://html.com/attributes/a-target/#Values_of_the_target_Attribute.

[4]   P. J. P. D. D. N. Nitin Naik, "2016 IEEE International Conference on Computer and Information Technology," *Native Web Communcations Protocols and Their Effects on the Perfomance of Web Services and Systems.,* 2016.

[5]   "Performance Evaluation of Dynamic and Static WordPress-based Websites," IEEE, Croatia, 2019.

[6]   M. MacDonald, "Creating a Website: The Missing Manual, 4th Edition," O'Reilly Media, INC, 06 2015. [Online]. Available: https://www.oreilly.com/library/view/creating-a-website/9781449374525/ch01.html.

[7]   A. T. J. B. M. B. N. A. Utkarsh Dubey, "A Hybird Authentication System for Websites on Mobile Browsers," IEEE, Bangalore, 201.

[8]   H. Benayed, "Basic Authentication and Form-based Authentication using Spring Security," medium.com, 04 05 2020. [Online]. Available: https://medium.com/@haytambenayed/basic-authentication-and-form-based-authentication-using-spring-security-ed79951dbb2e.

[9]   "Firebase Documentation,"[Online]. Available: https://firebase.google.com/docs.

[10]   "W3Schools," [Online]. Available: https://www.w3schools.com/.