

**MICHIGAN STATE**  

---

**U N I V E R S I T Y**

# Project Plan

## AutoHook Editor

The Capstone Experience

Team Urban Science

Jon Stover  
Ben LaFleur  
Daiwei Zhang  
Jeff Fallon  
Zach Lewis

Department of Computer Science and Engineering  
Michigan State University

Fall 2019



*From Students...  
...to Professionals*

# Functional Specifications

---

- AutoHook's central services must go through a long, inefficient process to make HTML changes for their client's AutoHook templates.
- They are looking for an efficient way to create, modify, and export their web designs.
- Our project core functionality allows them to modify websites in browser by bringing everything they need into one convenient application.



# Design Specifications

---

- Landing page displaying their list of clients, selecting a client displays various projects.
- Once a project is selected, the web editing environment is opened, displaying the template on the left half of the screen with the HTML code on the right.
- After making changes, export the template to a relational database to be imported as needed later.



# Screen Mockup: Project Select

The screenshot shows a web browser window with the URL <https://www.driveautohook.com/autohooktemplateeditor>. The page features the "DRIVE AutoHook" logo, which is powered by Urban Science. There are three main interactive elements at the top: a green "Create" button, a green "Delete" button, and a search input field labeled "Search...".

Below these elements is a list of car brands on the left and a table of project entries on the right. The car brands listed are Acura, Alfa Romeo, Audi, BMW, FCA, Ford, GM, Honda, Hyundai, and Infiniti, with a downward-pointing triangle below Infiniti. The table has two columns: "Name:" and "Last Accessed:". The table contains 12 rows of project data, all with a "Last Accessed" date of 9/9/1755.

| Name:                        | Last Accessed: |
|------------------------------|----------------|
| acuraproject1print.html      | 9/9/1755       |
| acuraproject1redeem.html     | 9/9/1755       |
| acuraproject1submission.html | 9/9/1755       |
| acuraproject2print.html      | 9/9/1755       |
| acuraproject2redeem.html     | 9/9/1755       |
| acuraproject2submission.html | 9/9/1755       |
| acuraproject3print.html      | 9/9/1755       |
| acuraproject3redeem.html     | 9/9/1755       |
| acuraproject3submission.html | 9/9/1755       |
| acuraproject4print.html      | 9/9/1755       |
| acuraproject4redeem.html     | 9/9/1755       |
| acuraproject4submission.html | 9/9/1755       |
| acuraproject5print.html      | 9/9/1755       |
| acuraproject5redeem.html     | 9/9/1755       |
| acuraproject5submission.html | 9/9/1755       |
| acuraproject6print.html      | 9/9/1755       |
| acuraproject6redeem.html     | 9/9/1755       |

At the bottom right of the table area, there are three elements: a green "Last" button, a green "Next" button, and the text "1 out of 20 pages".



# Screen Mockup: Project Select Clicked

The screenshot shows a web browser window with the URL <https://www.driveautohook.com/autohooktemplateeditor>. The page header includes the logo for DRIVE AutoHook, powered by URBAN SCIENCE, and navigation buttons for 'Create' and 'Delete', along with a search bar containing 'Search...'. On the left side, there is a vertical list of car brands: Acura, Alfa Romeo, Audi, BMW, FCA, Ford, GM, Honda (highlighted in green), Hyundai, and Infiniti, with a downward arrow below. The main content area is a table with two columns: 'Name:' and 'Last Accessed:'. The table contains six rows of data, all with a 'Last Accessed' date of 9/9/1755. The first row is 'hondaproject1print.html', the second is 'hondaproject1redeem.html', the third is 'hondaproject1submission.html', the fourth is 'hondaproject2print.html', the fifth is 'hondaproject2redeem.html', and the sixth is 'hondaproject2submission.html'. At the bottom right of the table, there are 'Last' and 'Next' buttons and a page indicator '1 out of 1 pages'.

| Name:                        | Last Accessed: |
|------------------------------|----------------|
| hondaproject1print.html      | 9/9/1755       |
| hondaproject1redeem.html     | 9/9/1755       |
| hondaproject1submission.html | 9/9/1755       |
| hondaproject2print.html      | 9/9/1755       |
| hondaproject2redeem.html     | 9/9/1755       |
| hondaproject2submission.html | 9/9/1755       |





# Screen Mockup: Project Select Delete

AutoHook Powered by Urban Science

DRIVE **AutoHook** POWERED BY URBAN SCIENCE

Create Delete Search...

| Name:                   | Last Accessed: |
|-------------------------|----------------|
| hondaproject1print.html | 9/9/1755       |
| hondaproject1re         | 9/9/1755       |
| hondaproject1s          | 9/9/1755       |
| hondaproject2p          | 9/9/1755       |
| hondaproject2r          | 9/9/1755       |
| hondaproject2s          | 9/9/1755       |

**ARE YOU SURE**

Delete "hondaproject2print.html"?

No Yes

**Acura**  
**Alfa Romeo**  
**Audi**  
**BMW**  
**FCA**  
**Ford**  
**GM**  
**Honda**  
**Hyundai**  
**Infiniti**

Last Next 1 out of 1 pages



# Screen Mockup: Template Editor

The screenshot displays the 'AutoHook Powered by Urban Science' template editor. The browser address bar shows the URL `https://www.driveautohook.com/autohooktemplateeditor`. The interface includes three green buttons: 'Back', 'Load', and 'Save'. The project title is 'Acura: Project 1'. The DRIVE AutoHook logo is visible in the top right corner.

The editor is split into two main sections:

- Left Section (Form Mockup):** A preview of a 'RECEIVE A \$50 REWARD CARD' form. The form is titled 'With Test Drive at Peter Testing Chrysler'. It contains two columns of input fields: 'First Name', 'Last Name', 'Email', 'Zip Code', and 'Phone Number'. Below these are dropdown menus for 'New or Used', 'Any Year', 'Any Make', and 'Any Model'. There are 'Get Your Code' and 'Not Now' buttons at the bottom.
- Right Section (Code Editor):** A text area showing the HTML code for the form. The code is numbered from 01 to 22 and includes DOCTYPE, HTML, HEAD, and BODY tags with various attributes and content.





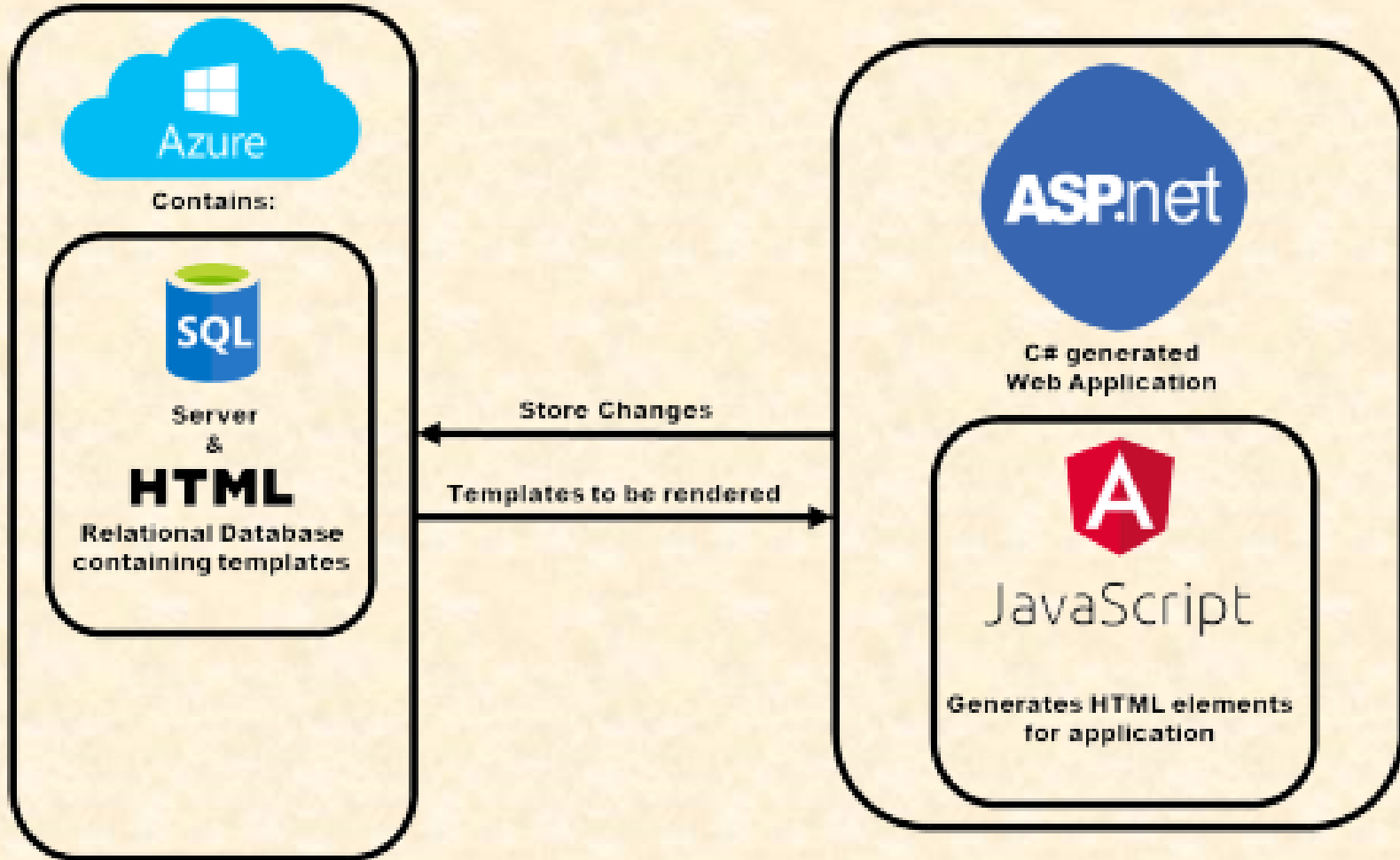
# Technical Specifications

---

- Our project is a data driven web application focused on modifying HTML through an intuitive UI
- Allows dynamic import/export of HTML/CSS template files using a relational database stored in the Microsoft Azure Cloud
- HTML editor displays live changes to code in iframe window



# System Architecture



# System Components

- Hardware Platforms
  - Microsoft Azure Cloud services
- Software Platforms / Technologies
  - Microsoft Visual Studio
    - ASP.net web development
    - JavaScript Angular framework front end
    - C# back end maintaining state information
  - Microsoft Azure Cloud
    - Relational database
    - Server



# Risks

- Importing and Exporting HTML:
  - Description: Importing and exporting web page templates will require our project to move HTML to and from a relational database.
  - Mitigation: Once we get our web application to display data from our server, we will be experimenting with selecting and inserting large amounts of HTML and CSS from a database. We can then figure out how much of this content is necessary to select and insert, allowing us to minimize the need for storage on our server, and making importing/exporting web templates faster and more efficient.
- Display the HTML inside of Web App with Live Dynamic Updates:
  - Description: Our clients want a core aspect of our web application to be direct modification of the HTML and CSS that generates their content, from within a web browser.
  - Mitigation: We have researched an open source solution called ASP.net AJAX Control Toolkit which contains a tool called HTML editor that we believe we can modify to perform a good part of the html modification part of our project, allowing us to focus on our drag and drop tools.
- Intuitive GUI:
  - Description: Our client has a large volume of the templates that they wish for our tool to act on.
  - Mitigation: We have already built a relational database containing fields which have some basic html in them. We have performed a select statement on this database from within our application. We will wrap statements to create new items and manage files in buttons.
- Undo functionality:
  - Description: An intuitive interface should include the ability to undo actions.
  - Mitigation: We have a relational database which contains multiple copies of the html which serves up the template we are working on. We have an algorithm for having multiple versions which you can call back to.



# Questions?

---

?

?

?

?

?

?

?

?

?

