MICHIGAN STATE UNIVERSITY

Project Plan Presentation

ViSUI: Video Simplified User Interface

The Capstone Experience

Team TechSmith

Jered Brophy
Scott Isaacson
Averi Justice
Jack Koby
Diego Marzejon
Erika Zheng

Department of Computer Science and Engineering
Michigan State University

Spring 2022

From Students…
…to Professionals

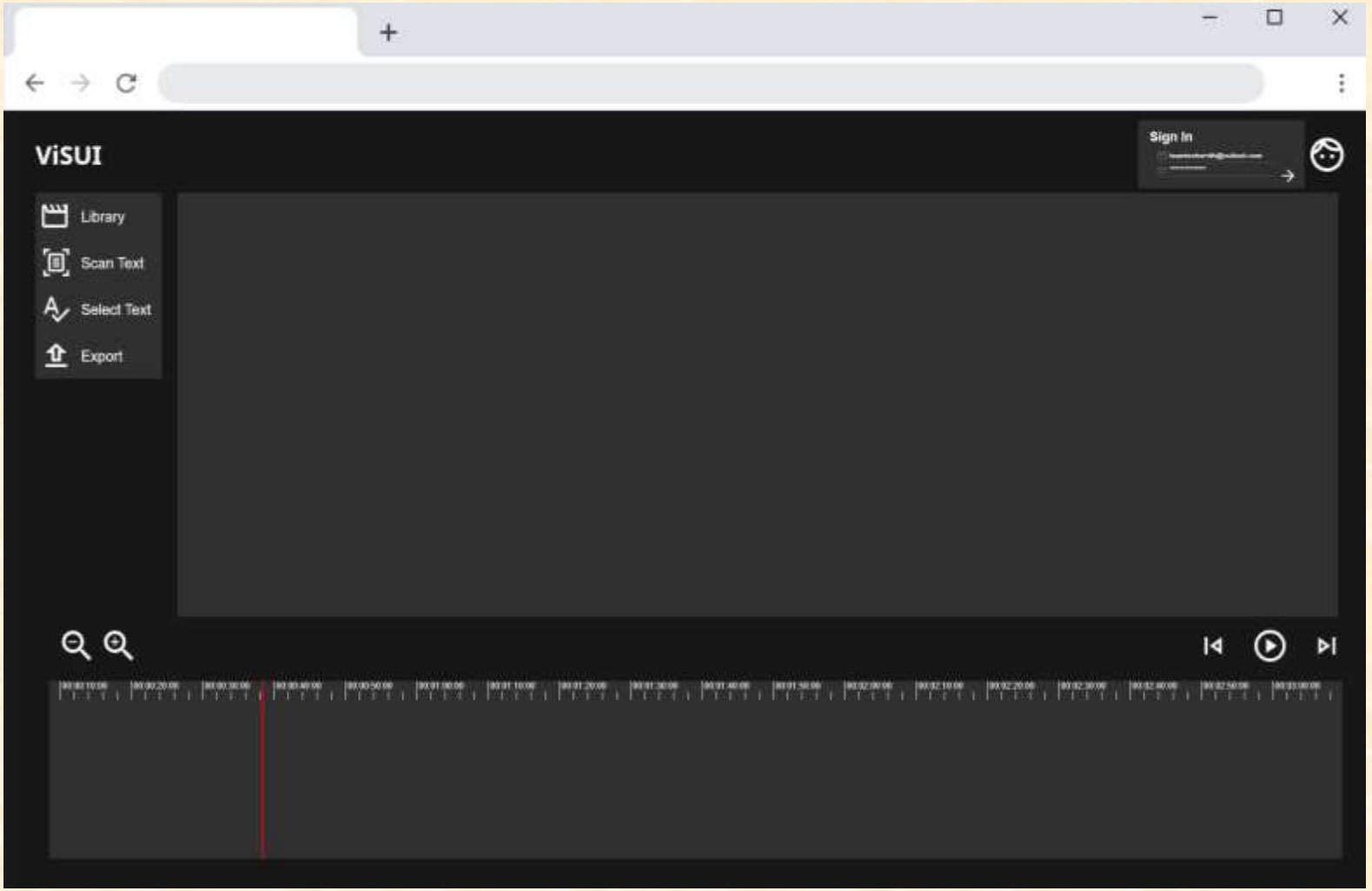
Functional Specifications

- ViSUI Web Application to help simplify video content
  - Blocking unneeded textual content
  - Create more language agnostic demos
- This project solves the problem of additional subject matter interfering with the ability to understand the content a video is trying to convey
- Unneeded content or on-screen pop ups can cause distraction and can pull away user's attention
- ViSUI removes these distractions by identifying their location within the video and suggesting fixes for the user to choose from. These changes include blocking or blurring the unwanted content
- The user can then save and export this edited video file

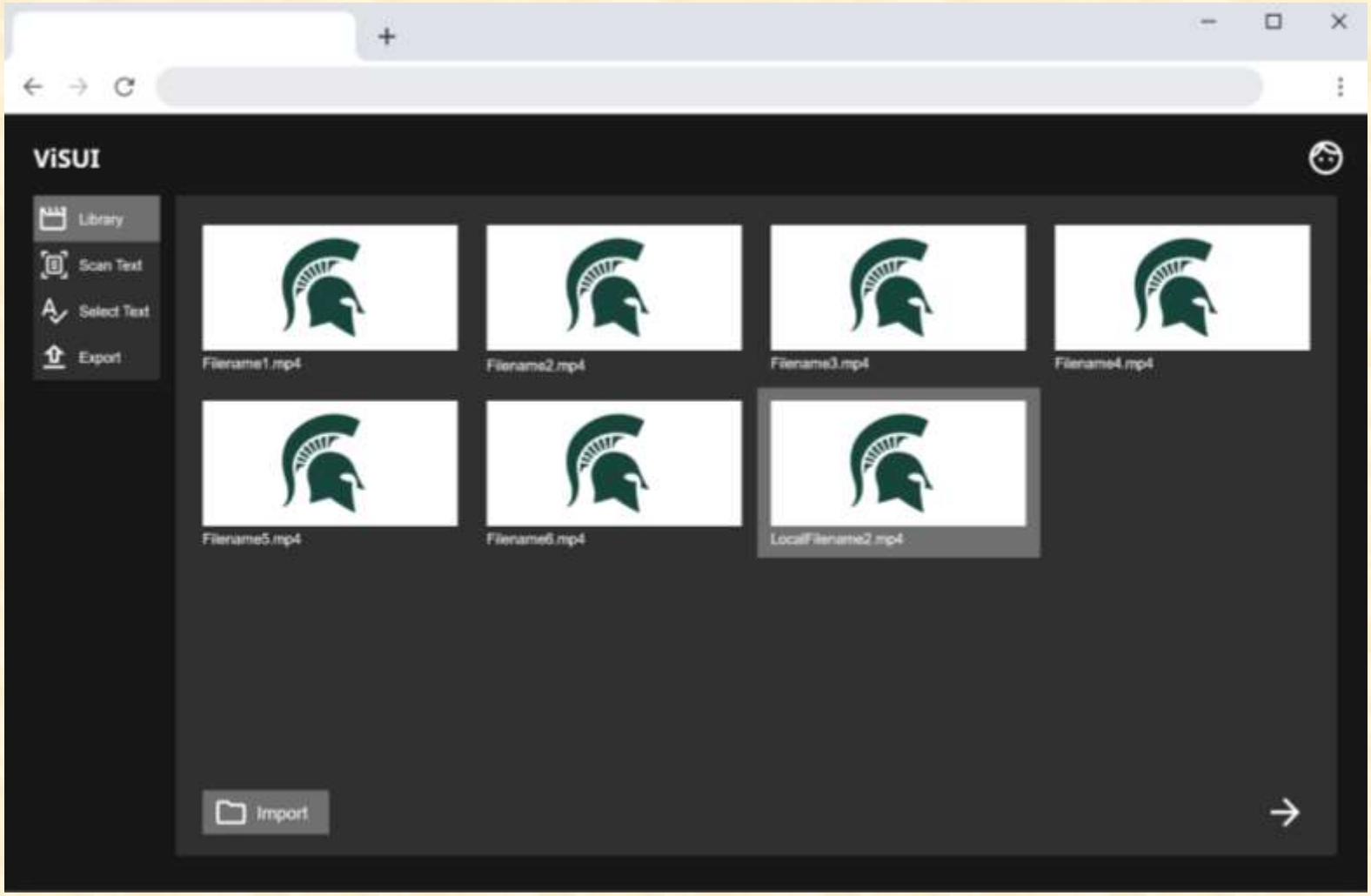
Design Specifications

- Web application for TechSmith customers
- Users can import, edit, save, and export videos
- Area for suggestions made to simplify text in video
- Text scanning and text selecting buttons
- Ability to adjust size of SUI rectangles
- Video scrubbing tool that can be zoomed in and out for precise sampling
- Button for removing audio
- Users have access to a library collection of all videos they've uploaded

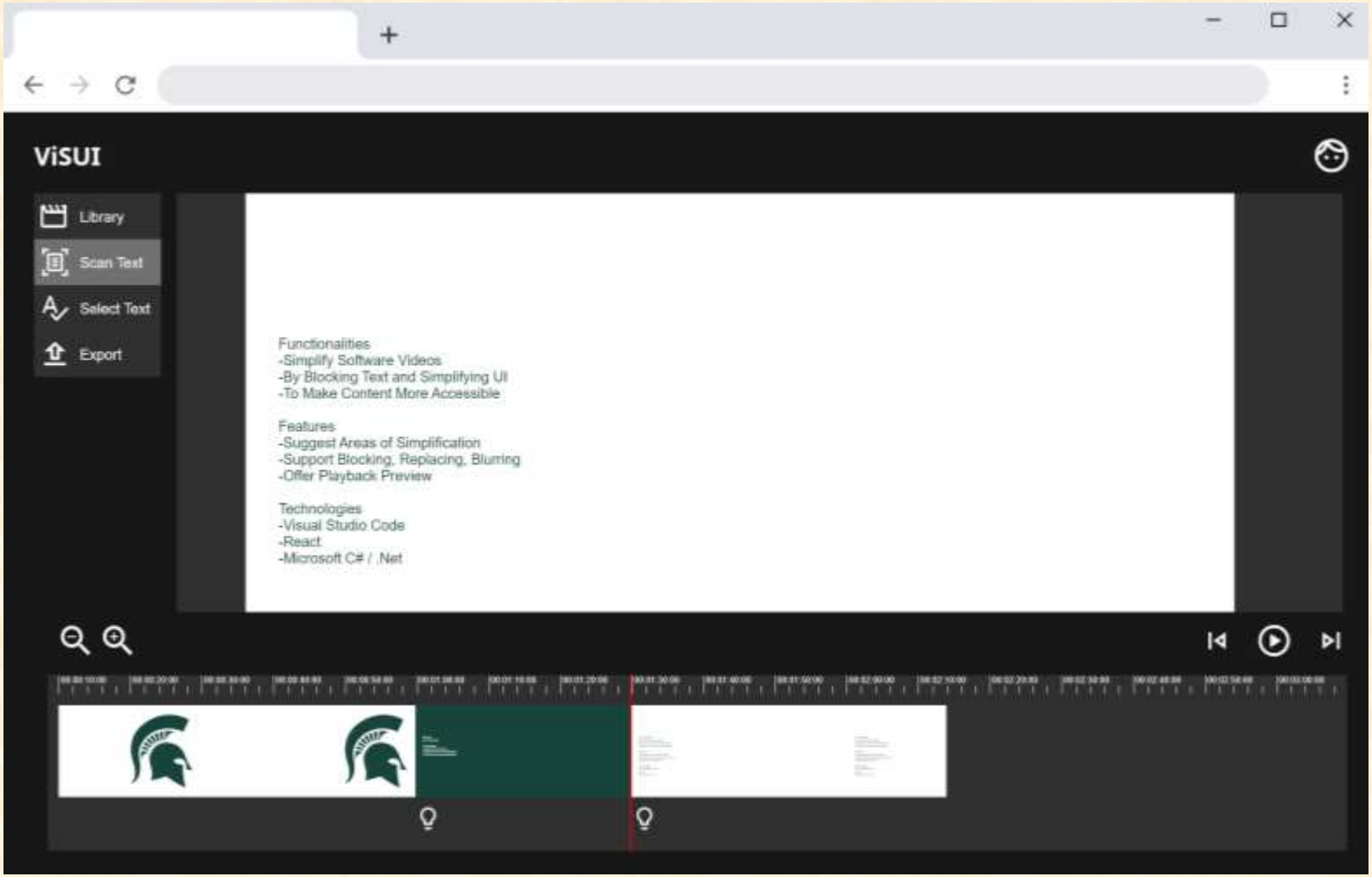
Screen Mockup: Home Page



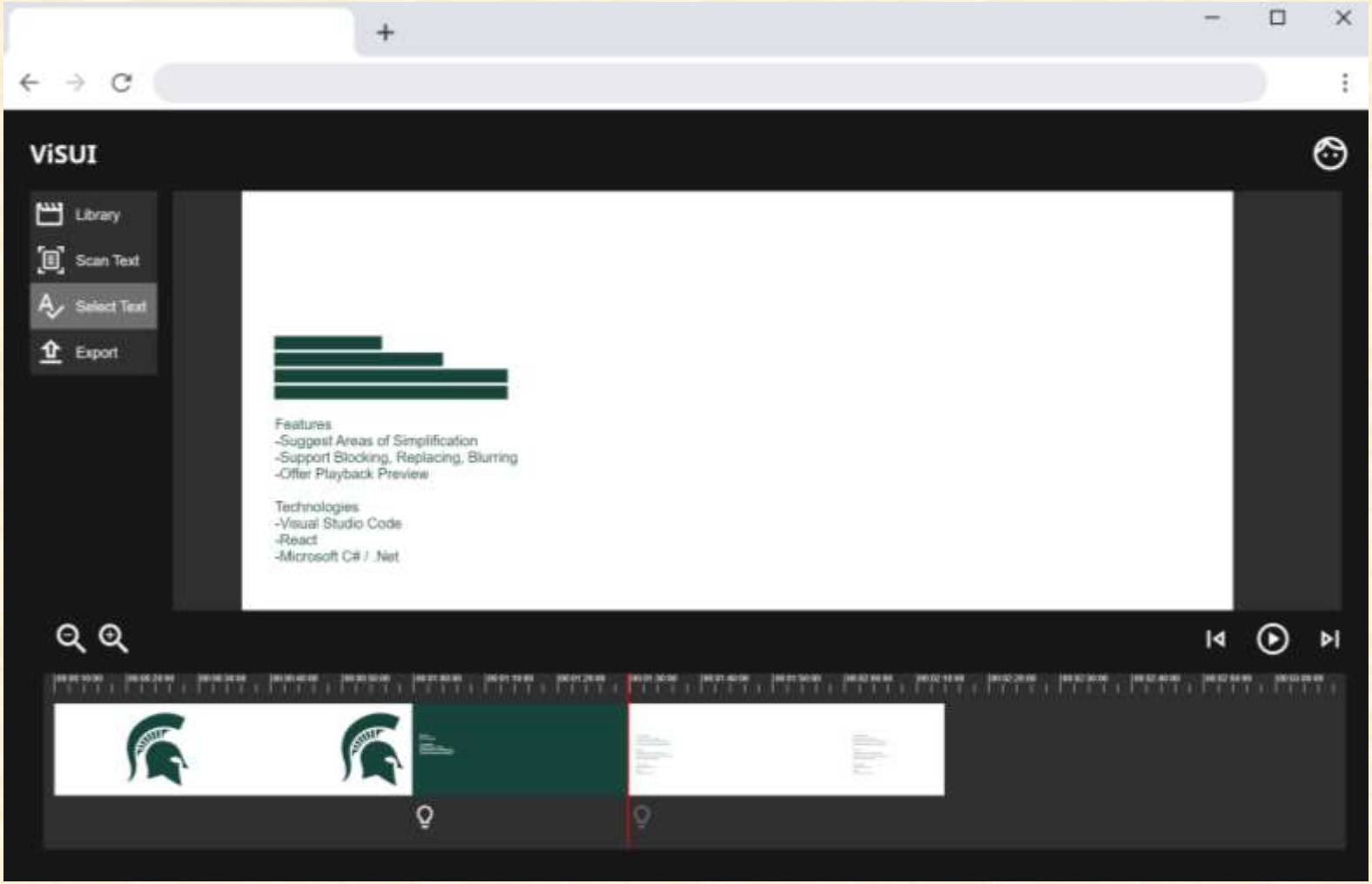
Screen Mockup: Video Library



Screen Mockup: Scan and Suggest



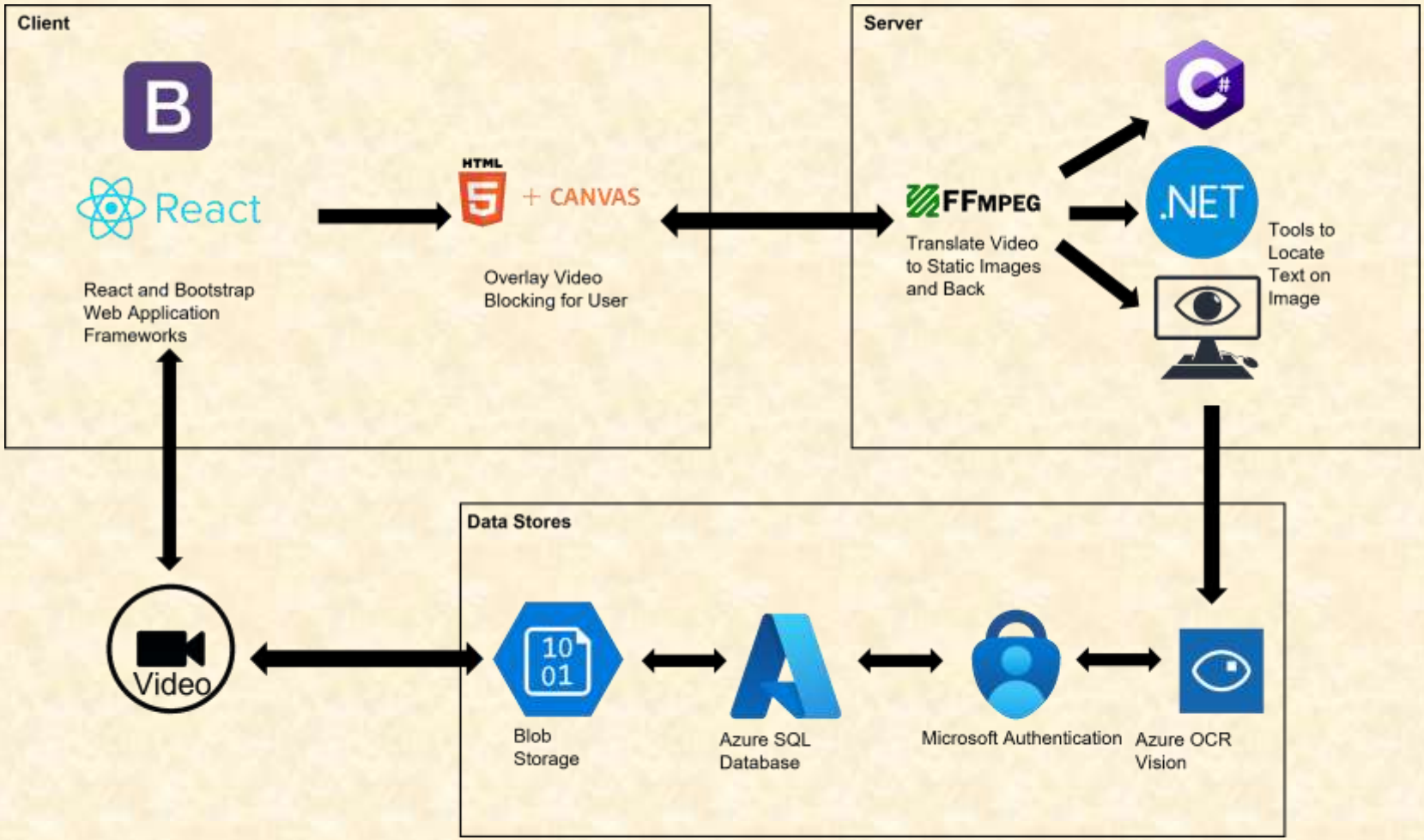
Screen Mockup: Text Blocking



Technical Specifications

- React
- C#/.Net
- Microsoft's Computer Vision
- Microsoft Azure Blob Storage
- FFmpeg

System Architecture



System Components

- Hardware Platforms
  - Server running on Azure services
- Software Platforms / Technologies
  - Microsoft Computer Vision's Optical Character Recognition
  - FFmpeg Video Software
  - Microsoft Azure Blob Storage
  - React, C#/.Net

Risks

- Risk 1
  - **Description:** When sampling, where to create breaks for suggestions and edits. Many routes to go with this, must agree on a design.
  - **Mitigation:** We believe that we could take snippets of video and scan these rather than scanning every individual frame. Must determine a process in which "good" snippets can be extracted and used without poorly effecting efficiency.
- Risk 2
  - **Description:** Efficiency of the sampling process, must find the “sweet spot” of how many images we want to analyze using the Computer Vision service.
  - **Mitigation:** Want to break up frames into groups that are large enough to increase efficiency but small enough to catch all changes. We expect to use our recently constructed code which plots points to create a box around where text appears to help determine good frames to pull.
- Risk 3
  - **Description:** Storage, updating, and exporting of the database. Making sure that we can save video files with our added modifications at any point during production.
  - **Mitigation:** Potentially use state changes to determine when a file is edited so that we know when to update in the database.
- Risk 4
  - **Description:** Determining how we want to modify the original video uploaded. Try to maintain efficiency without using more storage space than needed.
  - **Mitigation:** We plan to store the original video, determine what changes need to be made, and plug in modified frames into original video. (Rather than create an entirely new video after storing the original).

Questions?