

Red Cloud Indian School
Lakhóta Immersion Classroom

BOOMBRIDGE

PROJECT DOCUMENTATION

Carnegie Mellon University
Caitlin Huang, Kevin Wang, and David Yuan

TABLE OF CONTENTS

Introduction

The Problem	3
What is BoomBridge?	3
About the Developers	4

Setup

Deployment	5
Configuring the Canvas Plug-in	6
Creating Assignments	8

Maintenance

Software Stack	10
Code Structure	11
Emergency Response Plans	13

A Couple More Things

FERPA Compliance	15
Data Protection & Privacy	15

INTRODUCTION

The Problem

When we were first contacted by the Lakhóta Immersion Classroom program, we were tasked to deliver a product that does all of the following:

- **Self-Directed Learning:** allow students to work through courses at their own pace.
- **Progress Tracking:** make it easy for teachers to track the progress of students.
- **Interactive Learning:** deliver lectures material to students in an interactive way

We evaluated the technology stack used by the Lakhóta Immersion Classroom program and realized that Canvas and BoomCards complete the set of features needed. While Canvas offers self-directly learning and progress tracking capabilities, it can be complemented by the interactive lessons offered by BoomCards. Unfortunately, there was no solution that allowed Canvas to communicate with Boom Cards directly.

What is BoomBridge?

We created BoomBridge to help address the disconnection between Canvas and BoomCards. It is a Canvas plug-in that processes student-submitted BoomCards screenshots and automatically grades the corresponding Canvas assignments.

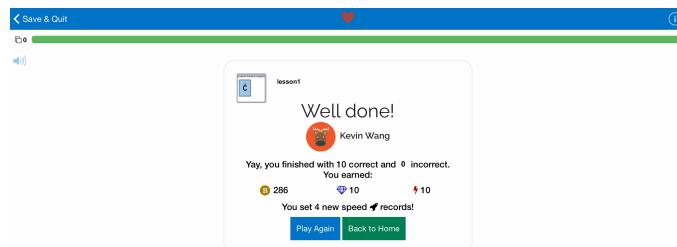


Figure 1: BoomCards Result Page Screenshot

First, the teacher would create Canvas assignments using BoomBridge as an external tool for each BoomCards deck. The student would complete the BoomCards deck using the link posted on the

Canvas assignment pages and submit a screenshot of their BoomCards result page once they are done with a lecture (see Figure 1). Then, the BoomBridge uses optical character recognition (OCR) to determine a grade from the screenshot, and automatically grades the assignment as well as unlocks the next lectures on Canvas.

About the Developers

Caitlin Huang

Caitlin was the Quality Assurance Lead. She is a third-year Information Systems student with an additional major in Statistics and Machine Learning. She will intern with UnitedHealth Group this summer and Finisar this fall. In her spare time, Caitlin enjoys running, trading, baking, and playing with her dog.

Kevin Wang

Kevin was the Project Manager and Design Lead. He is a third-year Information Systems student with an additional major in Human-Computer Interaction, graduating in May 2022. This summer, he will be interning with Apple as an Engineering Program Manager Intern. In his spare time, Kevin enjoys making apps, gaming, running, and playing lacrosse.

David Yuan

David was the Client Relations Management Lead. He is a fourth-year Information Systems student with an additional major in Statistics and Machine Learning, graduating in May 2021. He is working full-time at Tapad starting this summer. In his spare time, David enjoys playing basketball, cooking, and sampling exotic foods.

SETUP

Deployment

We have opted to deliver BoomBridge via the following ways:

1. CMU-IS Server

We understand that deploying an application at Red Cloud's servers requires leadership approval.

As a temporary strategy, we have deployed the BoomBridge application using a server instance (CMU-IS server) graciously provided by the CMU Information Systems department. The server can be reached at <http://darkknight-cmuis.net>.

This sandbox server is available for free to Red Cloud for up to three months after the conclusion of this project. During these three months, if for any reason the Immersion Classroom program decides to move this app to be hosted on Red Cloud's on-premise servers, the CMU IS department will gladly provide an ISO image for an easy transition. The ISO image can be used by Red Cloud and its vendors to create virtual machine instances that are identical to the CMU IS server.

After the three-month trial period has concluded, we will gladly migrate the server to either Red Cloud's servers using an ISO image or transfer the ownership of the existing server to our Red Cloud.

For information regarding BoomBridge deployed to the CMU-IS server, please contact Prof. Larry Heimann at profh@cmu.edu or (412)-268-3259.

2. Red Cloud On-Premise Server

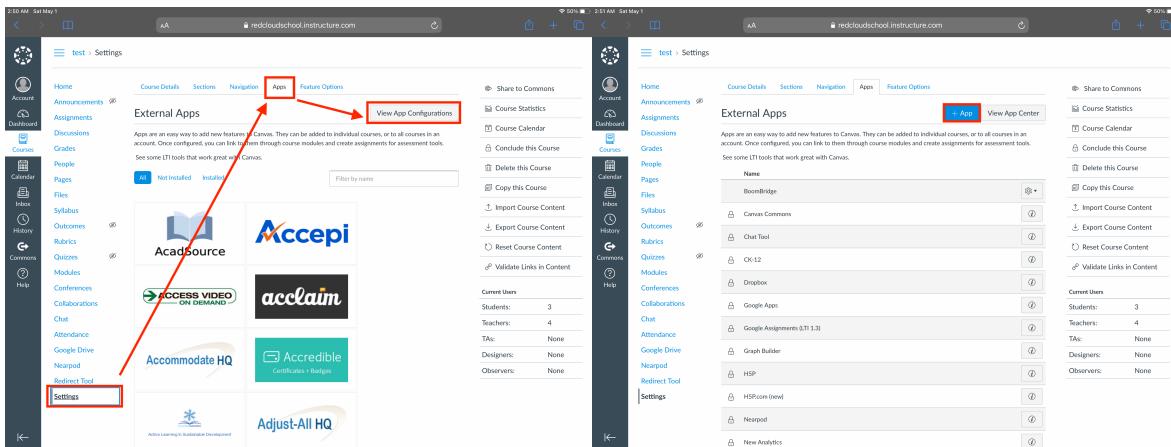
In this delivery option, we hope to deploy this application to Red Cloud's on-premise servers, given approval from Red Cloud's leadership team. The following are the required technical specification for the server instance:

Specs Category	Minimum	Recommended
Memory (RAM)	2GB	4GB
CPU	1 vCPU	2 vCPU
Disk Space	20GB	40GB
Server Operation System	Any Linux	Ubuntu
Server Software	Any	Nginx
Software Stack	Ruby on Rails v2.7.2	Ruby on Rails v2.7.2
Virtual Machine	Virtual machine is acceptable	Virtual machine is acceptable

Configuring the Canvas Plug-in

We designed this tool to work with any learning management tool that supports Learning Tools Interoperability (LTI). The tool is optimized for Canvas as it is the current learning management system used at Red Cloud. BoomBridge needs to be configured for each Canvas course. The following are instructions for loading BoomBridge as an External App in Canvas.

1. Go to your Canvas course and click on Settings > Apps > View App Configurations. Then click on "+App" to add a new External App.



2. Fill in the Name, Consumer Key, and Shared Secret as shown below. Set the Privacy to Public.
 - a. You should change the consumer key and shared secret if you have changed them on the server's "config/lti_settings.yml" file. You can read more about this in "Code Structure" in the next section.

- b. If you have migrated from the CMU-hosted server to your on-premise server, you should use the domain pointing to that server for the launch URL. For example, if your domain name is apple.com, then your new launch URL should be "<https://apple.com/launch>"
- c. You must set the privacy setting to Public.

The screenshot shows the 'Edit App' dialog box in a web browser. The dialog has the following fields:

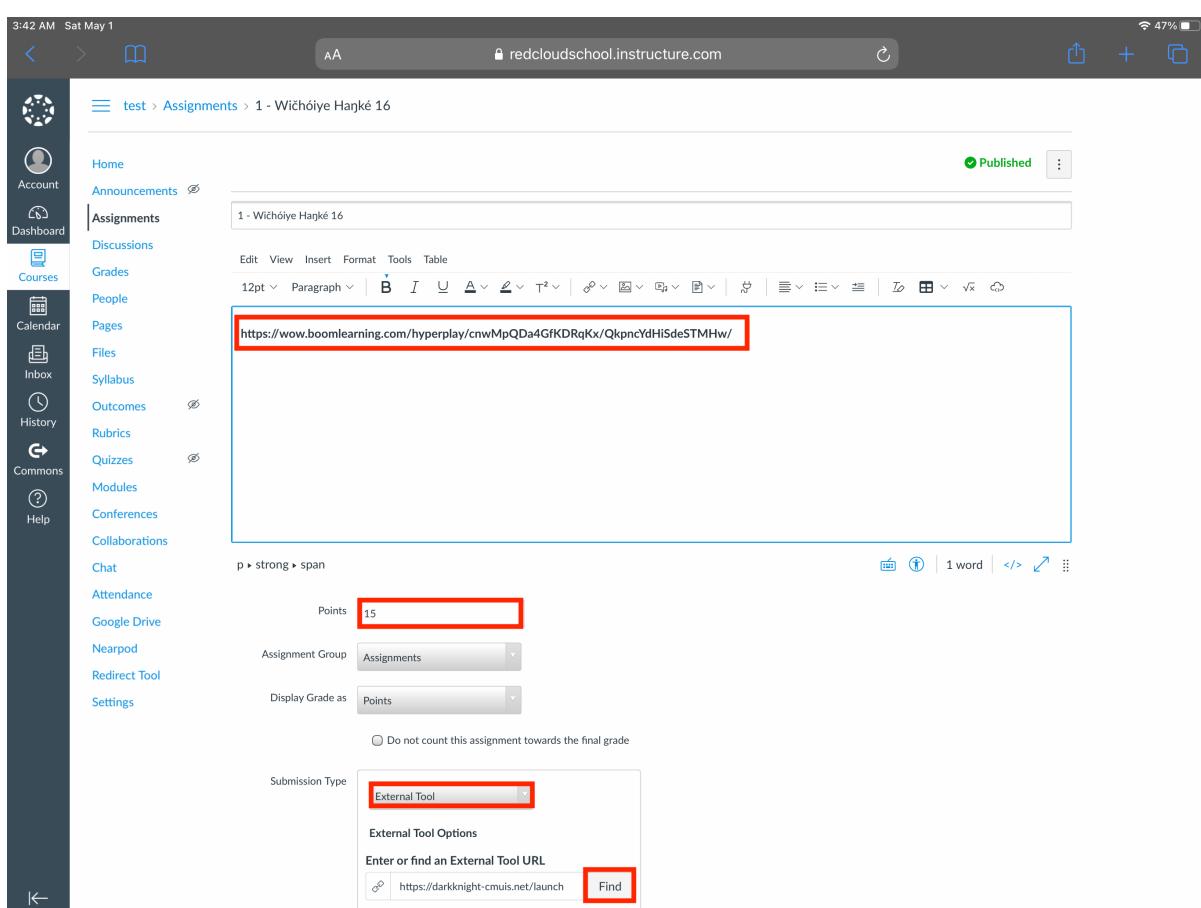
- Name:** BoomBridge
- Consumer Key:** ckckck
- Shared Secret:** cscscs
- Launch URL:** https://darkknight-cmuis.net/launch
- Domain:** Domain
- Privacy:** Public
- Custom Fields:** (Empty)

At the bottom of the dialog are 'Cancel' and 'Submit' buttons, with 'Submit' being highlighted by a red box.

3. Finally, click "Submit".

Creating Assignments

1. Get a link to a Boom Cards deck, and post it in a new assignment on Canvas. This link should not require student sign-in.
2. Change the points to the total possible points in the Boom Cards deck.
⚠ The total possible score or “points” is different from the number of slides in the deck.



The screenshot shows the Canvas LMS interface. On the left is a dark sidebar with various navigation links: Account, Dashboard, Courses, Calendar, Inbox, History, Commons, and Help. The main content area shows a course titled "test > Assignments > 1 - Wičhóye Härké 16". The assignment is titled "1 - Wičhóye Härké 16". The assignment details include:

- Points: 15 (highlighted with a red box)
- Assignment Group: Assignments (highlighted with a red box)
- Display Grade as: Points
- Submission Type: External Tool (highlighted with a red box)
- External Tool Options: Enter or find an External Tool URL (highlighted with a red box) containing the URL "https://darkknight-cmuis.net/launch" and a Find button (highlighted with a red box).

3. Configure the submission type to external tool, and click on “Find” to load BoomBridge as the external tool.
4. Save and publish
⚠ Note that it is normal to receive an authentication error upon creating the assignment. To preview BoomBridge, you should always be in student view.
5. Go to modules.
6. Add the assignment to a new or existing module.
7. Click on the “more” button, then “edit”, to bring up the module panel.
8. Configure the minimum score to unlock the next assignment. Add prerequisites if necessary.

3:37 AM Sat May 1

redcloudschool.instructure.com

test > Modules

Home Announcements ⚡ Assignments Discussions Grades People Courses Calendar Inbox History Commons Help

Account

Dashboard

Modules Conferences Collaborations Chat Attendance Google Drive Nearpod Redirect Tool Settings

Module 1

- Lesson 1 10 pts | Score at least 10.0
- Lesson 2 10 pts | Score at least 10.0
- Lesson 3 10 pts | Score at least 10.0

Wólyuthe

- 1 - Wičhóiyé Hajké 16 15 pts | Score at least 12.0
- 2 - Wičhóiyé Hajké 15 15 pts | Score at least 12.0
- 3 - wičhóiyé Hajké 14 10 pts | Score at least 7.0

Complete All Items + ⋮

Move Contents... Move Module... Delete Send To... Copy To... Share to Commons Commons Favorites

Edit

The screenshot shows the RedCloud Schools Instructure interface. The top navigation bar includes icons for back, forward, search, and refresh, along with the URL "redcloudschools.instructure.com". On the left, a sidebar lists various navigation items such as Home, Announcements, Assignments, Discussions, Grades, People, Pages, Files, Syllabus, Outcomes, Rubrics, Quizzes, Modules, Conferences, Collaborations, Chat, Attendance, Google Drive, Nearpod, Redirect Tool, and Settings. The main content area shows a module titled "Wōiyuthe" with sections for Prerequisites and Requirements. The Requirements section is expanded, showing three requirements: "1 - Wíčhóye Hąjké 1€", "2 - Wíčhóye Hąjké 15", and "3 - wičhóye Hąjké 14". Each requirement has a dropdown menu for "score at least", a score input field (12.0, 12.0, 7.0), and a progress bar indicating 15/15 for the first two and 10/10 for the third. A modal window titled "Edit Module Settings" is open over the main content, containing fields for the module title and prerequisites, and a requirements editor with the three listed requirements.

Edit Module Settings

Wōiyuthe

Lock until

Prerequisites

+ Add prerequisite

Requirements

Students must complete all of these requirements

Students must move through requirements in sequential order

Student must complete one of these requirements

1 - Wíčhóye Hąjké 1€	score at least	12.0	/ 15
2 - Wíčhóye Hąjké 15	score at least	12.0	/ 15
3 - wičhóye Hąjké 14	score at least	7.0	/ 10

MAINTENANCE

Software Stack

We use a stack of software frameworks to create BoomBridge. The maintainer should either have experience with these frameworks or contract third-party maintainers who have experience with these frameworks. The following is a non-exhaustive list that illustrates how we use crucial software frameworks to deliver the BoomBridge experience. You can access the BoomBridge code repository using the following link:

<https://github.com/sirkevinwang/BoomBridge-LTI>

If you need to gain access to the CMU-hosted server, refer to the “setup” section for the contact information of CMU’s server administrator.

Nginx

At the very bottom layer is Nginx, a web server that serves the BoomBridge web plug-in. The server we have deployed and the disk image that we will provide for the migration has Nginx preconfigured to handle web requests. Note that when using a disk image, a new SSL/TLS certificate must be given the domain change. We recommend using Let’s Encrypt which will come pre-installed as a part of the disk image.

Ruby on Rails

This application is built with Ruby on Rails 6, a popular framework used to build server-side web applications. We utilize Rails for handling simple requests to BoomBridge, such as rendering the web page that students see, establishing an LTI handshake with Canvas, and passing the grades back to Canvas via that handshake. We do not utilize the built-in ActiveRecord or ActiveMailer, which are used for database management and email management. We have excluded ActiveRecord and ActiveMailer purposefully.

Javascript + JQuery

Javascript is handled by Rails. We have written extensive Javascript code to do the following:

- Render user interface elements
- Send and receive AJAX requests to the Rails server
- Load student-submitted screenshot
- Perform on-device Optical Character Recognition (OCR) on student-submitted screenshots

JQuery ships with Rails 6 by default. We use JQuery to manipulate HTML elements more easily. Note that JQuery code is usually marked by a dollar sign ("\$") followed by a CSS selector.

Tesseract.js

From the very beginning, we designed this application around user privacy. When we ask students to submit their screenshots of Boom Cards lectures, we process that data on-device. We *never* send image data to our Rails server or any other web service. To be able to extract the grades from these screenshots, we chose to use Tesseract.js which allows us to perform OCR in-browser using Javascript. We then use the extracted text to determine the number of correct and incorrect questions answered and pass that information back to Canvas.

IMS-LTI Gem

BoomBridge uses the IMS-LTI gem, by Canvas. The Gem allows our app to quickly authenticate with Canvas using the IMS LTI standards. We also use this Gem to pass back grades through an encrypted channel.

Code Structure

In this section, we detail how we structured our code. We will highlight the key functions of each file and note certain considerations. We group the files we cover with their parent directories.

app/assets/images/

Under this directory are the image assets we use. For example, the icons on the home screen are stored here. These assets are compiled with the Rails 6 asset pipeline for optimized performance.

app/assets/javascripts/

- **application.js**: this is where we import the necessary Javascript libraries. Modify this file only when change the libraries used for this application
- **main.js**: the majority of the Javascript code is stored here. Since Javascript code is exposed to web browsers, we did not put extensive comments in this file. Hence, we document this file here.
 - > The preferred language is maintained using the `lakhotia` variable. An event listener is attached to the translate button and will change the relevant classes on text elements to toggle between languages.
 - > The `process_screenshot` function is called when a file is uploaded. First, Tesseract is used to extract text from the screenshot. Then, using a custom set of rule, we extract the

grade information from text. We first pass back that information to our server using the `fetch` method.

- > All the functions with the prefix `render`, `hide`, and `show` modify the views.
- > The `removeBlanks` function is not used anywhere as it has not been tested. It removes white spaces from image data to accelerate the OCR process.
- > There is an event listener attached to the `card-button`, which reset the views after a student fails to achieve full marks to allow them to make another submission.

app/assets/stylesheets

Files under this folder are CSS (or rather, SCSS) stylesheets used to style the web views.

app/controllers/application_controller.rb

This is the controller code for the entire BoomBridge application. In the `launch` method, we first authenticate with Canvas to retrieve student and assignment information. We then set the `session` hashes to help us render the views. The `grade` method passes grades back to Canvas using a consumer token that is also observed in the `launch` method. It also renders a JSON response depending on the status of the grade pass-back.

app/views/

This folder contains the files for the views.

- `index.html.erb`: this page shows when one visits the base URL (i.e. the domain). It gives the visitor instruction on how to set up BoomBridge with Canvas.
- `launch_error.html.erb`: this page shows when there is any authentication error.
- `launch.html.erb`: this page renders the content on the launch URL. The view guides students to upload their screenshots and provides grade feedback.
⚠ Only the launch URL should be used on Canvas. Refer to the section where we cover the specifics on routes.rb.
- `privacy.html.erb`: this page contains the privacy policy and FERPA compliance warnings.

config/lti_settings.yml

This file contains the key and secret we use to encrypt communication with Canvas. The key and secret are exactly those you enter into Canvas when adding BoomBridge as a plug-in. You can easily change the consumer key and secret under the production code block. You should always restart the server to propagate these changes.

config/environments/production.rb

We use the built-in whitelist feature that ships with Rails 6. If there is a domain change, you should add the domain to this trusted list using the `config.hosts << "domain"` syntax within this file. For a change to this file to propagate, you should always restart the server.

⚠️ When migrating this application from one server to another, pay special attention to whether or not if there is a domain change as you might need to change this file.

Gemfile

This contains the ruby packages (Gems) and their respective versions. One should always run `bundle install` under the project directory prior to starting the server.

⚠️ Note that, generally, if a file is not covered by this section, it is not a file we have created or modified. Hence, you should avoid making edits to these files.

Emergency Response Plans

While we have performed rigorous testing on BoomBridge, we want to provide you with a list of the potential problems you might encounter as well as the response plans to them. If there are multiple solutions lists, please consider following them sequentially.

Problem	Solution
On Canvas, the embedded assignment appears as blank	<ol style="list-style-type: none">1. Check the server and see if it is up and running.2. Confirm that the LTI tool is properly configured in the Canvas course settings (e.g. URL, consumer key and secret).3. Check Canvas system status page.
The gift unwrapping page hangs	<ol style="list-style-type: none">1. Check that the screenshot shows the correct and incorrect question count.2. Check if Boom Cards has modified their user interface on the results page. If that is the case, then the rules in <code>main.js</code> file will need to be modified.3. If you are testing as a teacher, this is expected. You should always test an assignment under the student view.
Canvas reporting grades that are not whole numbers.	<ol style="list-style-type: none">1. Ensure that the total points assigned on Canvas match the total points possible on Boom Cards.

Canvas not reporting grades	<ol style="list-style-type: none"> 1. Check to see if the screenshot is in the correct format. 2. Check to see if the server is still running. 3. Check Canvas system status page. 4. Check if the LTI tool is configured correctly with the right credentials. 5. Check server logs and see if there is an error message.
Canvas reporting incorrect grades	<ol style="list-style-type: none"> 1. Check if the total points configured for the assignment match the total points possible on Boom Cards. 2. Confirm that screenshot format has not changed. If it has changed, consider rewriting the rules for grade extraction in main.js.
Boom Bridge giving incorrect grades after submission	<ol style="list-style-type: none"> 1. Check screenshot format. 2. Confirm that screenshot format has not changed. If it has changed, consider rewriting the rules for grade extraction in main.js.
Canvas showing "Authentication Error"	<ol style="list-style-type: none"> 1. It is likely that you are in the teacher view. Enter student view to see a preview of the interface. 2. Check your consumer key and secret in Canvas matches that of BoomBridge.
"Errno::EACCES in Application#launch" Error	<ol style="list-style-type: none"> 1. You should first do <code>su rails</code> to enter the rails user account. Then you should navigate to <code>/home</code> and run <code>sudo chmod 777 -R BoomBridge</code> to give permission to server to access the folder. The password for rails can be found among the initial present upon logging in as root.

A COUPLE MORE THINGS

FERPA Compliance

The BoomBridge app depends on the usage of two external vendors: Boom Cards and Canvas. As both of those tools are individually FERPA compliant, BoomBridge, which does not collect or store any additional data, is also FERPA compliant. For more details, you can refer to the Boom Learning Privacy Policy (<https://wow.boomlearning.com/blog/privacy>) or the Canvas Instructure Privacy Policy (<https://www.instructure.com/policies/privacy>).

Data Protection & Privacy

BoomBridge collects no data. In other words, the application does not permanently store any data (except in memory to support user activities while they are interacting with this tool). However, we do process encrypted student and course information using the Canvas LTI APIs. Additionally, we process student screenshots on-device to extract grading information. We only send the extracted grades to our servers. Student uploaded screenshots are never sent to our servers and are kept strictly in the cache on their local machines.

User data is protected in the following ways:

- **Data Minimization:** we only collect the data we need to deliver the services provided by this tool.
- **On-Device Processing:** we use on-device machine learning to recognize grade information from screenshots. Student-submitted screenshots never leave their devices.
- **Control:** this LTI tool can be easily disconnected from Canvas.
- **Data Security:** student data is encrypted in both directions when we communicate with Canvas using a private key.