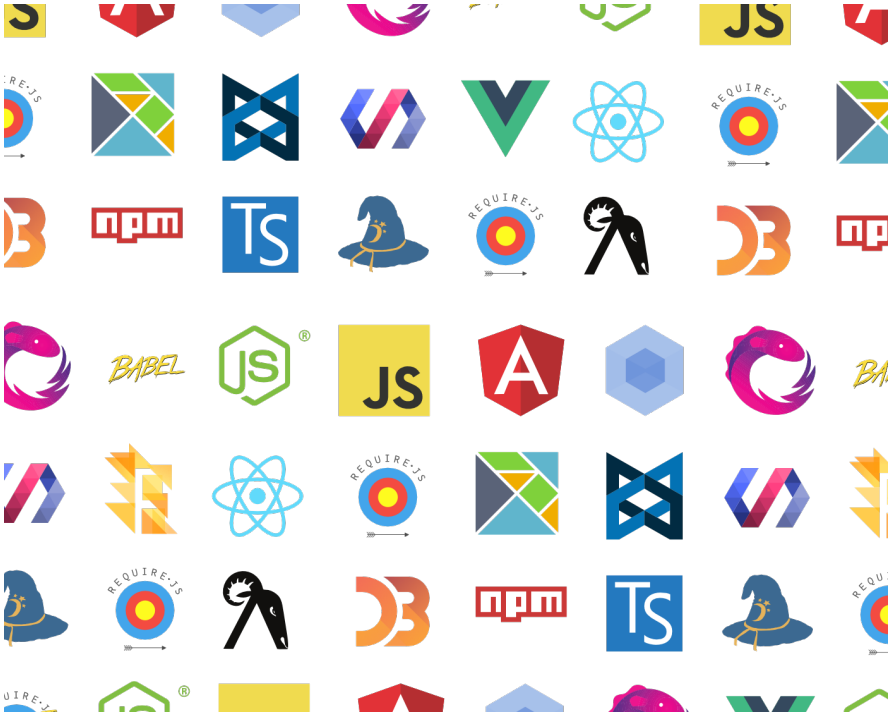
Taymaz Davoodi, M.Sc. Lecturer
tdavoodi@bu.edu

Office: (CDS-924)
Wed. 2PM-4PM
Fri. 5:15PM-7:15PM

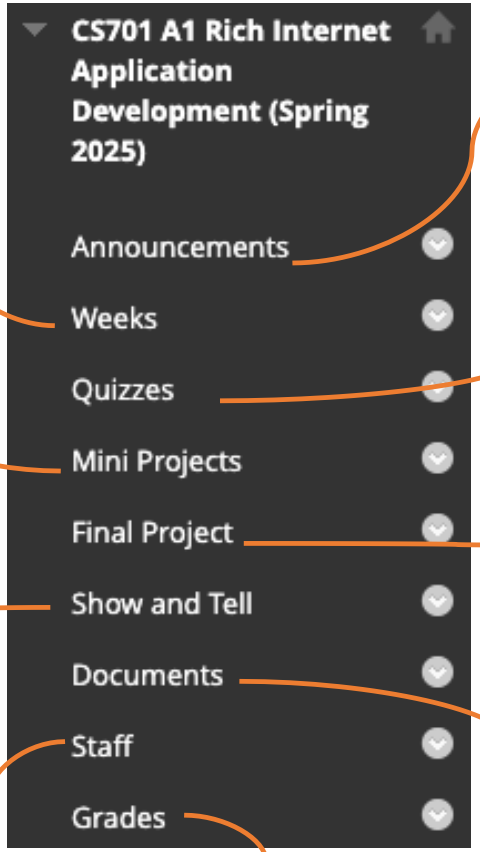
Welcome to MET-CS-701

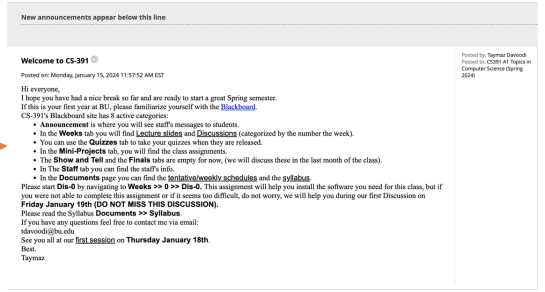
RIA Web-App Development

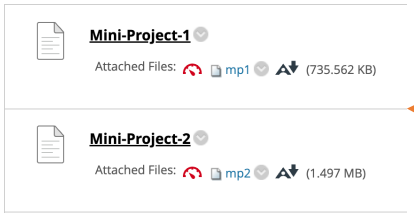


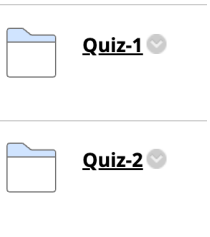
CS-701

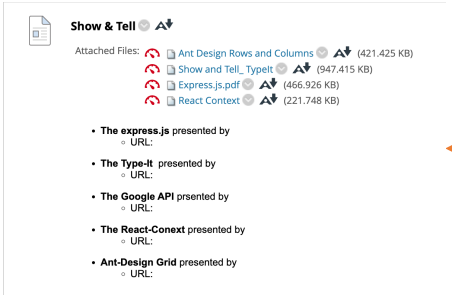
Blackboard

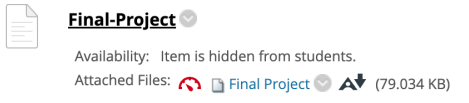


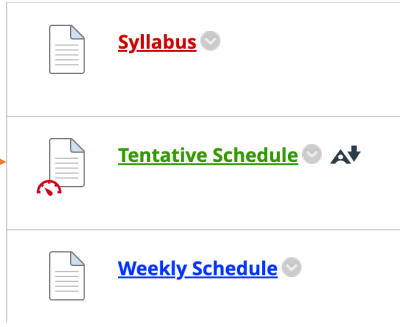


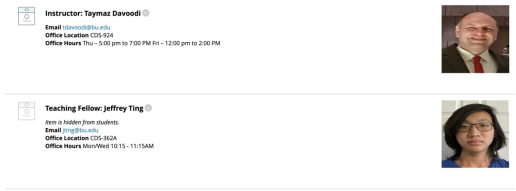


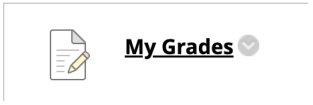












CS-701

IDE

Recommended:

WebStorm
Installation explained in Lab-0
(Free with BU email)

Acceptable:

VS Code

Will need SFTP extension or may need FileZilla

NOT Recommended:

Sublime

BB Edit

Notepad++

Course Material

Required

Set-up a **GitHub** account with your **BU-email**

Set-up a **Vercel** account and connect your GitHub account to it.

Gradescope for Assignment Submissions

Piazza for Class Communications

Blackboard for Assignment Submissions

Web

Recommended:

https://www.w3schools.com/

https://codepen.io/

https://html-css-js.com/

https://nextjs.org/docs

https://react.dev/reference/react

Prohibited:

CourseHero

Chegg

ChatGPT

CS-701

Grading

Weights (%)		Points
50	Quizzes (6)	50 x 5 = 250
10	Attendance	13-3=10
20	Mini-Projects (6)	20 x 5 = 100
20	Final Project	100
100		500

\+ 3% Show and Tell (Extra Credit)

\- 2 Lowest **Quiz** & **Lab** are dropped

\- 3 **Days** of absences are ignored (est.)

Collaboration

Prohibited:

- Copying all or part of someone else's work.
- Viewing all or part of someone else's work.
- Showing all or part of your work to another student.
- Consulting solutions from past semesters.
- posting your work where others can view it (e.g., online).

Collaboration Is Prohibited

(Unless Otherwise Stated)

(When) Permitted:

- Whom did you work with?
- What were their contributions?
- Where have you given them credit?

Attendance

Excused:

- Up to 3 days (est.)

Unexcused:

- (-1) point/day.

Acceptable Excuses (with documentation):

- Medical appointments/emergencies
- Faith-based holiday
- Job Interview
- Court order or Jury Service
- Death in Family

Unacceptable Excuses:

- Car broke down/traffic
- T/Metro/Bus broke down
- Forgot to set my alarm
- Missed flight
- Etc.

CS-701

Academic Misconduct

Cheating

- Cheating on an examination or assignment.
- Misrepresentation, falsification, or fabrication of data.
- Theft of an examination.
- Unauthorized communication during examinations.
- Knowingly allowing another student to represent your work as their own.
- Forgery, alteration, or knowing misuse of graded examinations, quizzes, grade lists, or official records or documents,
- Misrepresentation of identity
- Theft or destruction of examinations or papers
- Submitting substantially the same work in more than one course.

Plagiarism

Representing the work or ideas of another as your own; and/or using another's work or ideas without crediting the source.

Plagiarism includes, but is not limited to, the following:

- Copying the answers of another student on an examination.
- Copying or restating the work or ideas of another person or persons in any oral or written work (printed or electronic) without citing the appropriate source.
- Using audio or video footage that comes from another source (including work done by another student) without permission and/or acknowledgement of that source.
- Collaborating with someone else in an academic endeavor without acknowledging their contribution.
- Plagiarism can consist of acts of commission (appropriating the words or ideas of another as one's own), or omission (failing to acknowledge/document/credit the source or creator of words or ideas).

Consequences

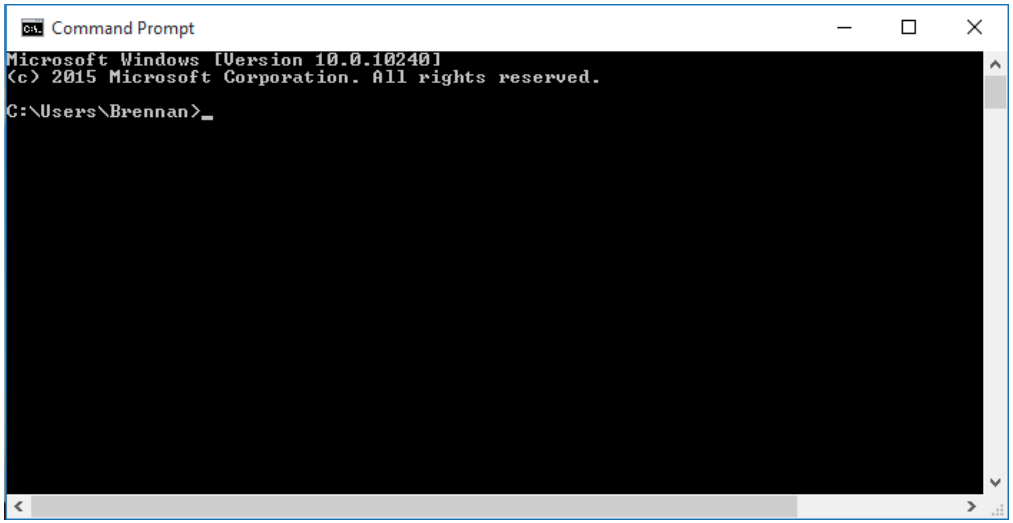
- Reprimand
- Disciplinary Probation
- Suspension
- Expulsion
- Revocation of Degree

CS-701

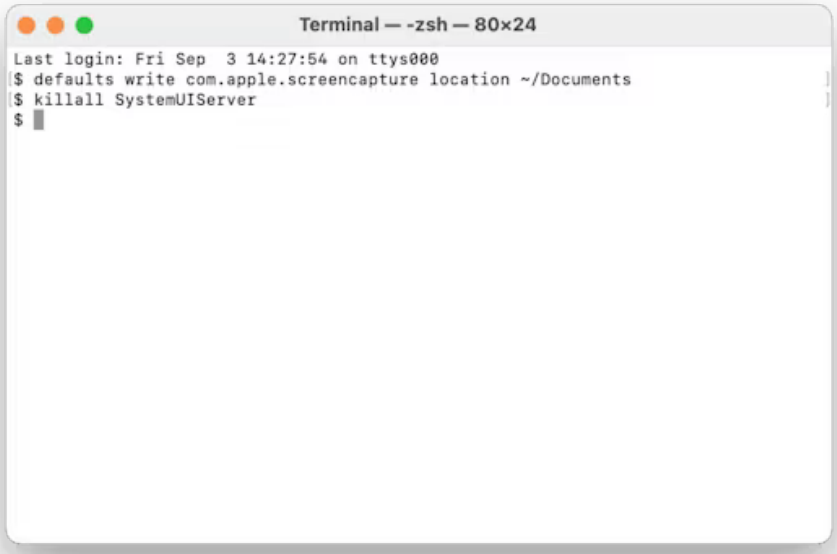
Command Line Interfaces (CLI)

In CS-391, we will be using the **CMD** (PC) / **Terminal** (MAC), frequently.

CMD



Terminal

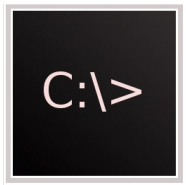


CS-701

Command Line Interfaces (CLI)

If you haven't had much experience with these **CLI**s, please familiarize yourself

CMD



Window Command	Action	Mac Command
Hit the "Windows" key -> Type in "cmd" -> Hit "Enter"	Open Terminal	cmd + spacebar -> Type in "terminal" -> Hit "Enter"
cd directoryWanted	Change Directory	cd directoryWanted
cd directoryWatned\SecondDirectory Wanted	Change Directory more than one step	cd directoryWanted/SecondDirectory Wanted **
cd ..	Step Back one directory(Parent Directory	cd ..
dir	View Content of Directory	ls
cls	Clear the Terminal	cmd + k
mkdir FolderName	Create New Directory	mkdir FolderName
rmdir FolderName *	Delete Empty Directory	rmdir FolderName *
rmdir /s FolderName	Delete Directory and its content	rm -R FolderName
cd.> FileName.type	Create New File	touch FileName.type
del FileName.txt	Delete File	rm FileName.type
cd\	Going Back to root	cd
Windows will find the first alphabetical match. Hit "tab" again to find the desired Directory/File	Auto-Complete Tab Button	Mac will require more information to distinguish which Directory/File is needed

Terminal

CS-701

All of your assignments (Mini-Projects) **must** be:

Deployed

Covered in Dis-0

Hosted

Vercel

Covered in Dis-1

Submitted

gradescope by Turnitin

Explained in MP-1

Note:
DON'T *UPDATE/PUSH/change* your assignment's GitHub repo after the **due-date**. **Altering your assignment after its due-date is considered cheating,** and at minimum, will result in an automatic 0 on that assignment.

You may change or alter that repository after you received a grade on that assignment.

CS-701

Deployment

There are different ways to deploy your assignments to GitHub. You can use a Git Management tool like GitHub-Desktop or you could just drag/drop your files to GitHub, but as CS-students you should know how to deploy your codes using **Terminal/CMD.**

First you have to download **Git** on your local machine

We Recommend

Git BASH

Included with Git download

Homebrew

Download: https://brew.sh/

After downloading and installing Git using an installer, you need to set your **Global Git configurations**

```
~ % git config --global user.name "Taymaz Davoodi"
~ % git config --global user.email "tdavoodi@bu.edu"
```

CS-701

Deployment

To connect your local-machine to GitHub, type the following commands in your Terminal/CMD

To be able to create a secure connection you should create a Secure Shell (**SSH**) **key**

```
~ % ssh-keygen
```

The Terminal/CMD will ask you a series of questions, to which you could just press ***enter***

```
Enter file in which to save the key (/Users/taymazdavoodi/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
```

The Terminal/CMD will automatically generate, and save an **SSH-Key**

```
Your identification has been saved in /Users/taymazdavoodi/.ssh/id_ed25519
Your public key has been saved in /Users/taymazdavoodi/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:CKv78TXRPKDQ7SD+vBsJ1R/6QdadvYmaAh0m03QYkYI taymazdavoodi@crc-dot1x-nat-10-239-151-251.bu.edu
```

If you closed your terminal you can always find your key again via terminal/CMD

Your public key has been saved in /Users/YOUR_USERNAME/.ssh/id_YOUR_ID.pub

To get to your key you need to display the content of id_YOUR_ID.pub file

```
taymazdavoodi@crc-dot1x-nat-10-239-151-251 .ssh % cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOcq5t4xzOdpiJq1nJ/g/VSHgc8ETjYymj/qQ+uFwcTJ
```

Copy **YOUR SSH-Key** to clipboard

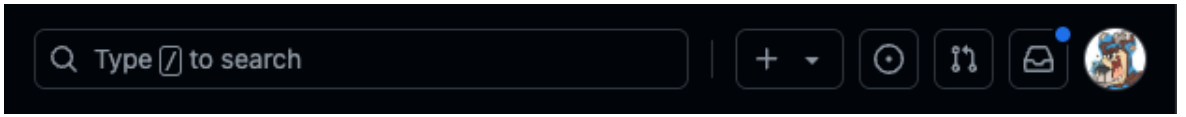
```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOcq5t4xzOdpiJq1nJ/g/VSHgc8ETjYymj/qQ+uFwcTJ
```

Don’t copy this one

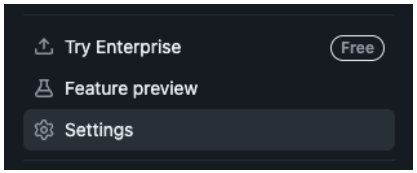
CS-701

Deployment

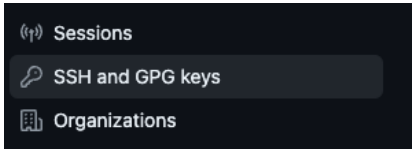
Sign-in to you account on GitHub, find your avatar (top-right cornet)



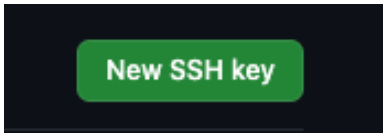
Click on your Avatar and navigate down to the Settings



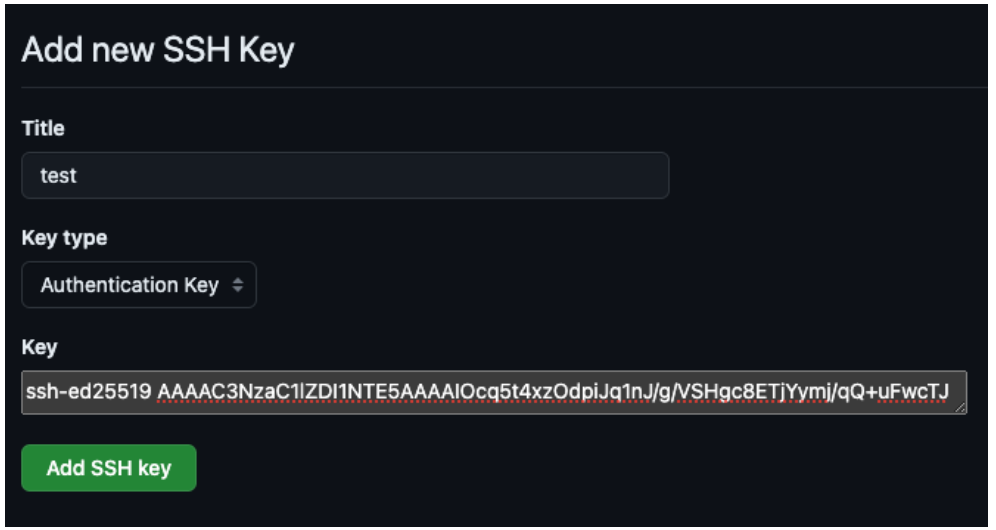
From the Settings page, navigate down to SSH and GPG keys



Find the New SSH key button and click on it.



Never share your Key



Add a Title for your local machine (e.g., Taymaz's Computer), paste YOUR **SSH-Key**, and click on Add SSH key

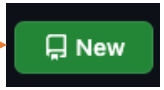
CS-701

Deployment

When you are done with each and every assignment (Mini-Projects), you **must** deploy that assignment to GitHub

```
~ % ls
~ % cd WebstormProjects
```

1. Create a repository on GitHub with matching names.
2. Locate the assignment on your local machine.
3. Enter the root directory of that assignment and initialize git there.
4. Add all files to your local repo.
5. Commit to a push with adding a comment.
6. Push all your files from your local machine to the repository on GitHub




```
% git init
```

```
% git add *
```

```
% git commit -m "initial-commit"
```

```
git remote add origin https://github.com/tazmanianDeviloper/demo.git
git branch -M main
git push -u origin main
```

You will repeat this process in **Dis-0** (Do Not Miss)

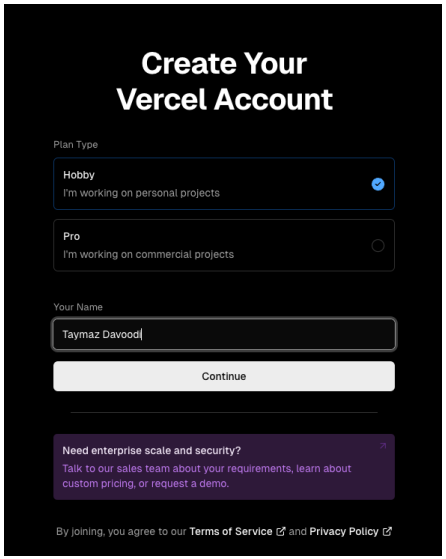
If done correctly, you should be able to see your assignment files on GitHub, (**Save the URL**)

CS-701

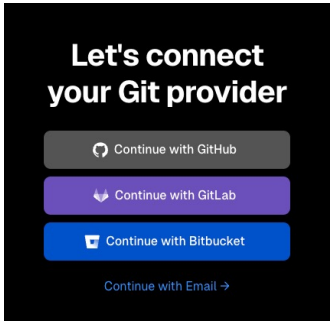
Hosting

Once your assignment files are on GitHub, you can have them hosted via **Vercel.**

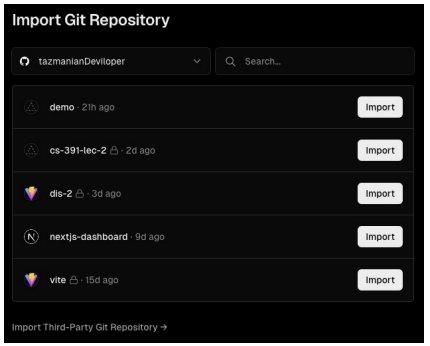
Setup a “**Hobby**” account



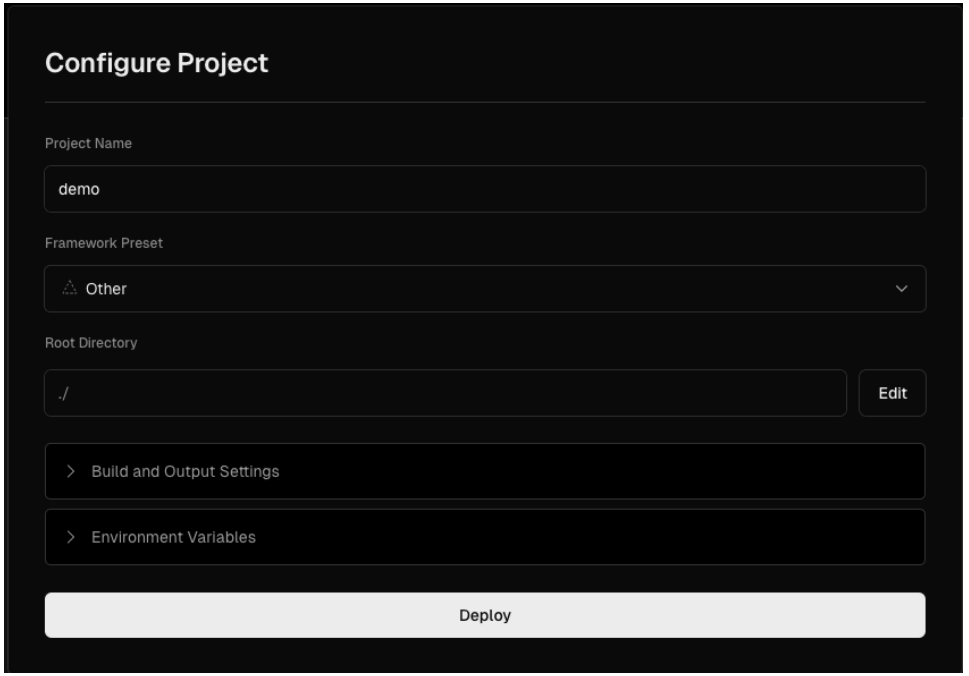
Connect GitHub to Vercel



Import your assignment’s repo



Usually, no additional configuration is needed, just click **Deploy**

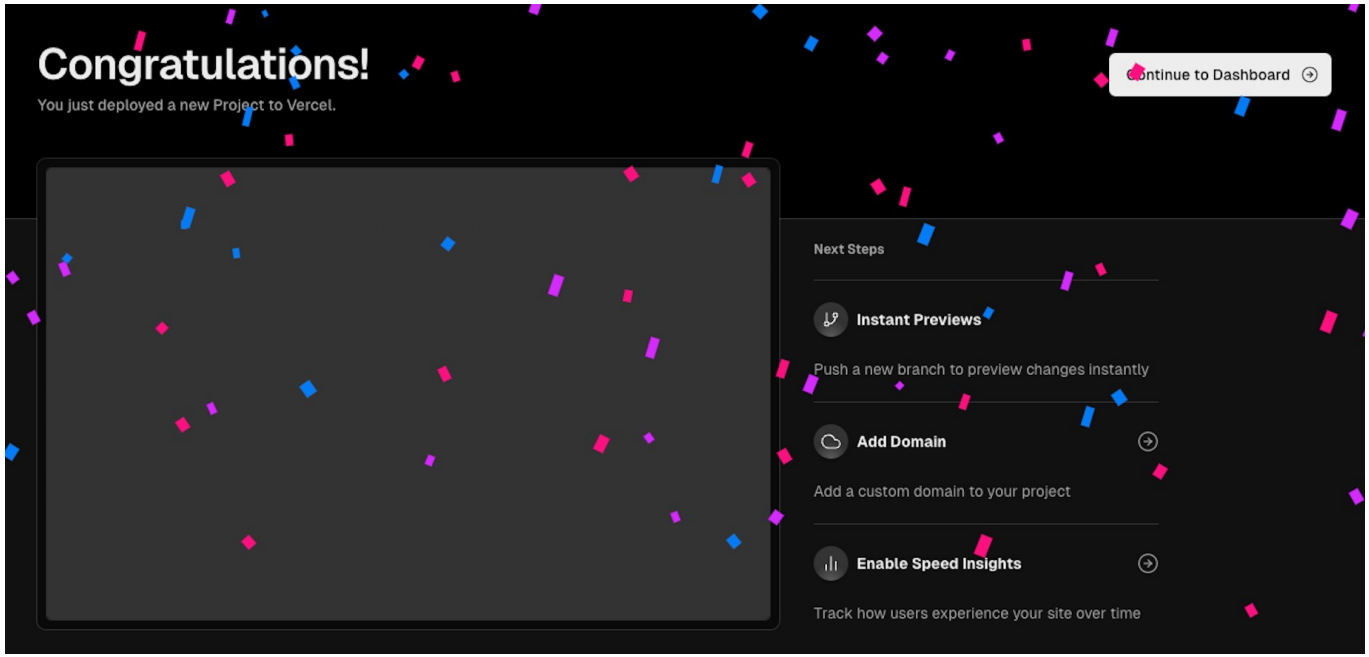


Vercel

CS-701

Hosting

Once you click Deploy, after a few seconds, Vercel will host your assignment on the internet, and prompt you with *Congratulations* message and a screen shot of your hosted assignment.



If done correctly, you should be able to see click on that screen shot, which would open a new browser tab to your assignment (**Save the URL**)

gradescope by Turnitin

CS-701

Submission

You have to submit both **URL**s (GitHub and Vercel) to the GradeScope

Create a simple textual file (.txt)

Paste both **URL**s (these are examples):

https://github.com/tazmanianDeviloper/demo
https://demo-xi-ten-35.vercel.app/

Convert to PDF and submit it to the GradeScope

Note:

- **Make sure your links lead to your assignment, if they don't graders will give you a zero (0) and there is no make-up**
- **Do not share your assignment URLs with other students; assignments must be completed individually.**
- **Do not push/update/change your code on GitHub after the due date of an ungraded assignment.**

CS-701

Terminal/CMD

Windows CMD	Task	Mac OS Terminal
`dir`	List files and folders	`ls`
`cd`	Full path of current folder/directory	`pwd`
`cd` <path to directory>	Change folder/directory	`cd` <path to directory>
`cd..`	One directory up in directory tree	`cd ..`
`cd`	Move to root directory	`cd /`
`mkdir newFolder`	Create new directory in current directory	`mkdir myFolder`
`echo some-text > fileName(.txt)`	Create new file	`cat > fileName(.txt)`
`rmdir myFolder`	Remove a directory*	`rmdir myFolder`
`ren oldFolderName newFolderName`	Rename a directory	`mv oldFolderName newFolderName`
`robocopy myFolder` <path to destination directory>	Copy a directory	`cp -r myFolder` <path to destination directory>
`move myFolder` <path to destination directory>	Move a directory	`mv myFolder` <path to destination directory>
`del myFile`	Remove a file*	`rm myFile`
`ren oldFileName newFileName`	Rename a file	`mv oldFileName newFileName`
`copy myFile` <path to destination directory>	Copy a file	`cp myFile` <path to destination directory>
`move myFile` <path to destination directory>	Move a file	`mv myFile` <path to destination directory>
`cls`	Clear the terminal screen	`clear`