

SE Deployment Workshop: *Extreme Startup*

17-313: Foundations of Software Engineering

<https://cmu-313.github.io>

Michael Hilton and **Josh Sunshine**

Spring 2026

Course Announcements

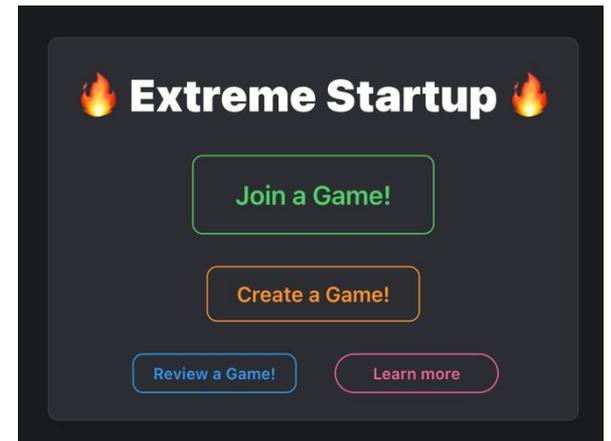
- Final Time:
 - Thursday, April 30, 2026 08:30am - 11:30am
 - Room TBD
- Midterm
 - Thursday, February 26
 - If you need a disability accommodation, please schedule with ODR immediately.
- Project 2C due ~~Thursday, February 26~~ Friday, February 27

Learning Goals

- Get familiar with continuous deployment
- Deal with fast-changing requirements based on user demand, which scales with success rate / user base
- Understand the value of in-house testing before deploying to production
- Differentiate between missing features and wrong behavior in a production system

Extreme Startup Game

- Maintained by Robert Chatley, Matt Wynne, and Mike Sorokin (Imperial College London)
- Goal is to help students take rapid deployment to the extreme
- We will play the game, then discuss as a group



Setup Instructions:

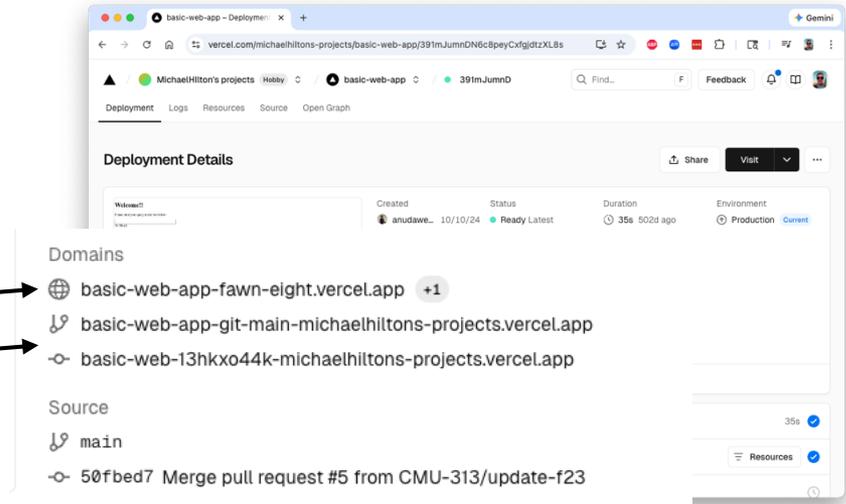
Step 1: have vercel app running

- If you have not done the recitation, you will need to do this first. (See course website)
- Visit your vercel deployment. Find the URL for your deployment

NOT this:

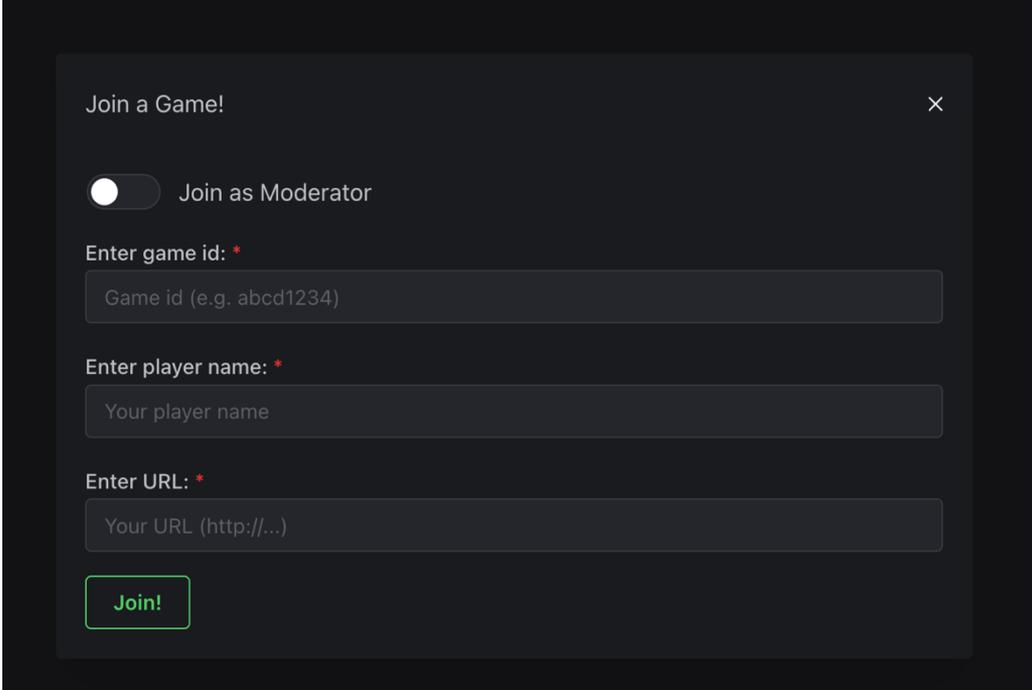
Use This:

NOT these:



Step 2: Join game

- Game id: from board
- Player Name (AndrewID):
 - Remember what you put in
- URL:
 - **IMPORTANT:** PASTE YOUR URL BUT ADD '/API'
 - E.g.
`http://myurl.com/API`



Join a Game! ×

Join as Moderator

Enter game id: *

Game id (e.g. abcd1234)

Enter player name: *

Your player name

Enter URL: *

Your URL (http://...)

Join!

Extreme Startup

Not Secure 128.2.24.106/7b268e54/players/1948b7c8

Player: hilton

Player ID

1948b7c8

Game ID

7b268e54 [Copy Game Id](#)

API

<https://basic-web-app-fawn-eight.vercel.app/api>

Score

-100

Events

ID	Query	Points	Timestamp	Outcome
3e751752	What is your name?	-10	01:09:31	INCORRECT
813aaeff	What is your name?	-10	01:09:26	INCORRECT

The screenshot shows the Vercel Logs interface for a project named "basic-web-app". The "Logs" tab is selected, and a search bar indicates "16 total logs found...". A blue arrow points from the "Logs" tab to the search bar. The left sidebar contains a "Filters" section with a "Timeline" dropdown set to "Past 30 minutes" and a "Contains Level" section with "Error 1" and "Warning 0" options. The main log table displays the following data:

Time	Status	Host	Request	Messages
FEB 24 20:09:57.42	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:52.17	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:46.79	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:41.96	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:36.93	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:31.69	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:26.37	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:21.22	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:16.08	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:10.74	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:05.41	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:09:00.26	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:08:54.91	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:08:49.57	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 20:08:43.28	GET 200	basic-web-ap...	/api	/api?q=What%20is%20your%20name%3F
FEB 24 19:59:28.61	GET 500	basic-web-ap...	/api	Node.js process exited with exit status: 1. Th

At the bottom of the log table, it says "No more logs to show within selected timeline".

See logs here

Takeaways

- Continuous deployment (auto-deploy on push) makes it easy to land changes in production
 - But there is always some delay in the feedback loop after your push
- Not having a feature is better than performing incorrectly
- Requirements can change quickly based on user demand
- User demand scales with success in production
- Local testing can help prevent issues in production, but testing in production might speed things up