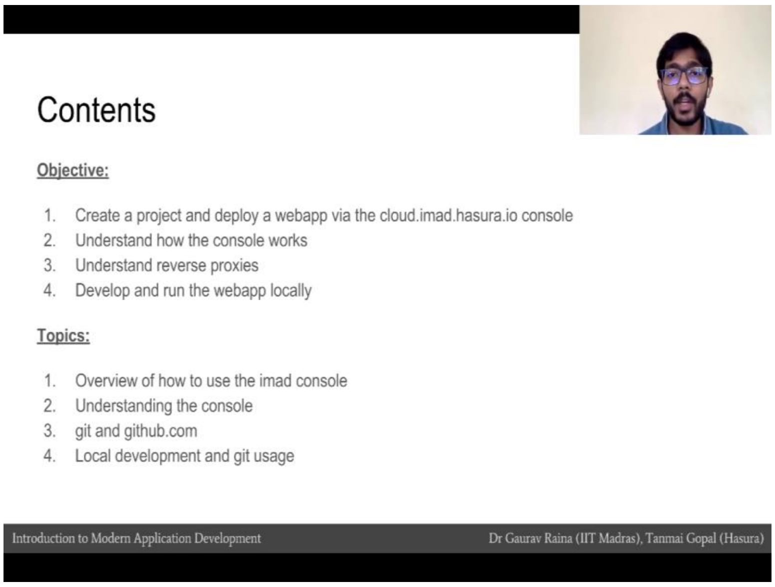
Introduction to Modern Application Development

Prof. Tanmai Gopal
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module - P3
Lecture - 09
Practical: Building a webapp with nodejs and using git for source code management
Introduction to Reverse Proxy

Hi everybody, welcome to module P3. In this practical module, we will be building a webapp in nodejs. And will also be using git for source code management. This module will also introduce us to the concept of a reverse proxy, which is a very interesting concept to understand, because it leads to things like load balances later on.

(Refer Slide Time: 00:18)



Contents

Objective:

1. Create a project and deploy a webapp via the cloud.imad.hasura.io console
2. Understand how the console works
3. Understand reverse proxies
4. Develop and run the webapp locally

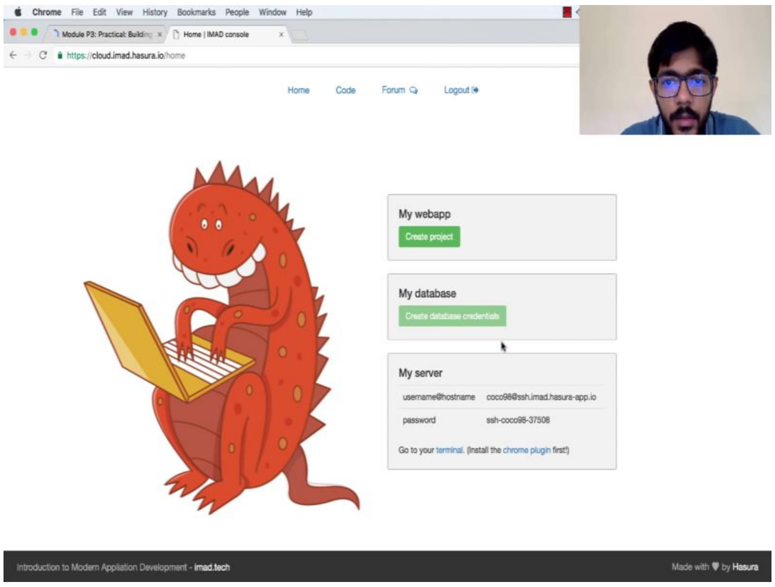
Topics:

1. Overview of how to use the imad console
2. Understanding the console
3. git and github.com
4. Local development and git usage

Introduction to Modern Application Development   Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

Our main objectives in this course are to create a project and deploy it online, and also to develop the webapp locally, and by locally I mean on your own computer system. We will also try to understand how deployment works, we have the cloud.imad.hasura.io console and that will lead us into a discussion about reverse proxies.

(Refer Slide Time: 00:41)


Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Home | IMAD console
https://cloud.imad.hasura.io/home
Home
Code
Forum
Logout
My webapp
Create project
My database
Create database credentials
My server

username@hostname	coco98@ssh.imad.hasura-app.io
password	ssh-coco98-37508

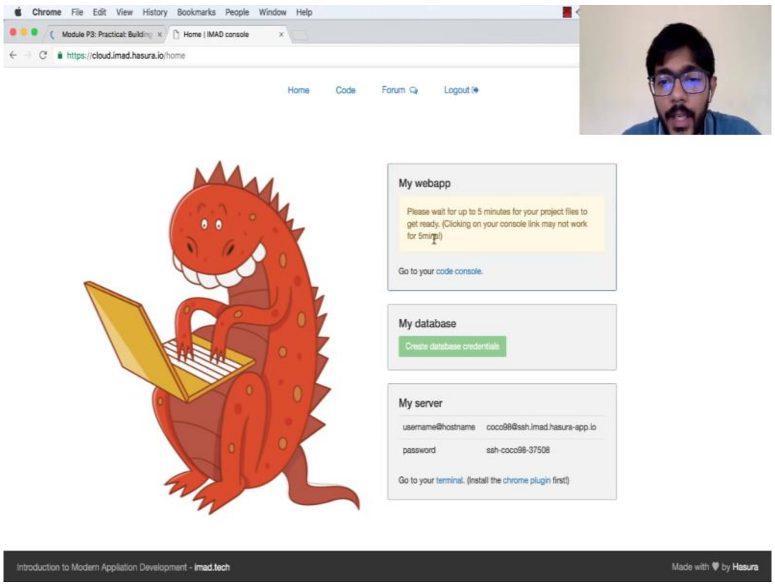
Go to your terminal. (Install the chrome plugin first!)

Introduction to Modern Appliation Development - imad.tech

Made with ♥ by Hasura

Let us head to the console. I go to my console. I can see that there is a button called create project, click on this button.

(Refer Slide Time: 00:47)


Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Home | IMAD console
https://cloud.imad.hasura.io/home
Home
Code
Forum
Logout
My webapp
My database
Create database credentials
My server

Please wait for up to 5 minutes for your project files to get ready. (Clicking on your console link may not work for 5mins!)

Go to your code console.

username@hostname	coco98@ssh.imad.hasura-app.io
password	ssh-coco98-37508

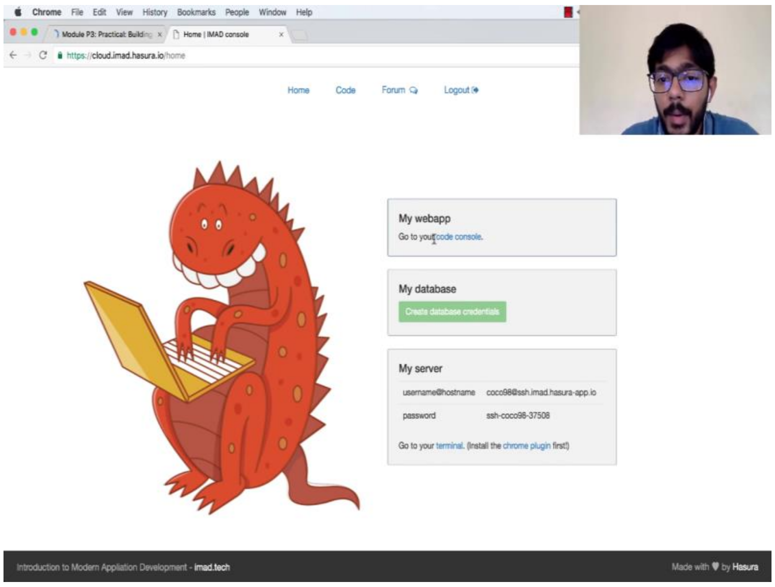
Go to your terminal. (Install the chrome plugin first!)

Introduction to Modern Appliation Development - imad.tech

Made with ♥ by Hasura

When I clicked on this button the source code for a basic webapp was copied

(Refer Slide Time: 00:49)


Home
Code
Forum
Logout
Create database credentials
Introduction to Modern Appliation Development - imad.tech
Made with by Hasura

My webapp

Go to your code console.

My database

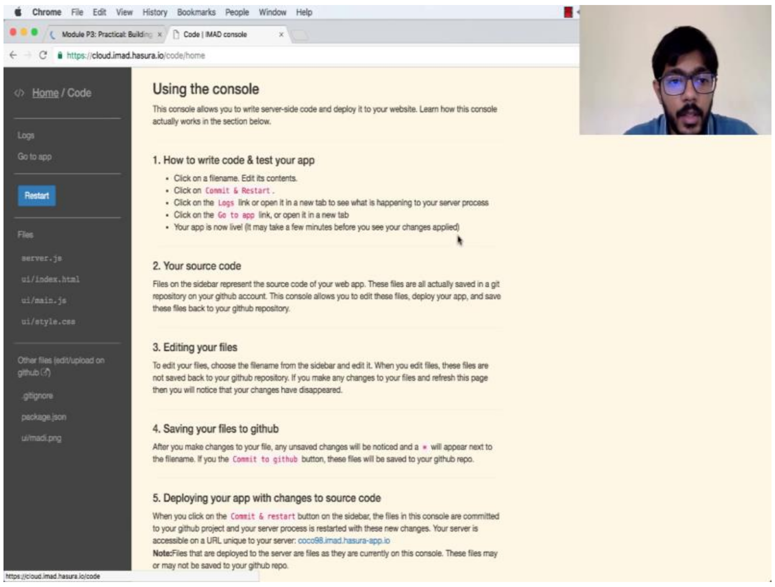
My server

username@hostname	coco98@ssh.imad.hasura-app.io
password	ssh-coco98-37508

Go to your terminal. (Install the chrome plugin first!)

And was created just from me on my github account and we will understand how this works later and that project was created and is now accessible on github, and also on the IMAD console. Now, I can go to the code console, will wait till all the files are loaded.

(Refer Slide Time: 01:09)



Using the console

This console allows you to write server-side code and deploy it to your website. Learn how this console actually works in the section below.

1. How to write code & test your app
   - Click on a filename. Edit its contents.
   - Click on `Commit & Restart`.
   - Click on the `Logs` link or open it in a new tab to see what is happening to your server process
   - Click on the `Go to app` link, or open it in a new tab
   - Your app is now live! (It may take a few minutes before you see your changes applied)

2. Your source code

Files on the sidebar represent the source code of your web app. These files are all actually saved in a git repository on your github account. This console allows you to edit these files, deploy your app, and save these files back to your github repository.

3. Editing your files

To edit your files, choose the filename from the sidebar and edit it. When you edit files, these files are not saved back to your github repository. If you make any changes to your files and refresh this page then you will notice that your changes have disappeared.

4. Saving your files to github

After you make changes to your file, any unsaved changes will be noticed and a * will appear next to the filename. If you the `Commit to github` button, these files will be saved to your github repo.

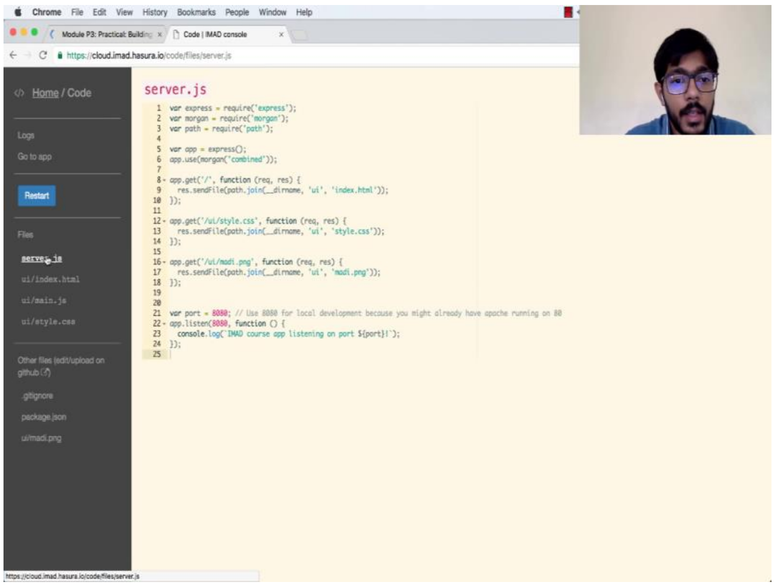
5. Deploying your app with changes to source code

When you click on the `Commit & restart` button on the sidebar, the files in this console are committed to your github project and your server process is restarted with these new changes. Your server is accessible on a URL unique to your server: coco98.imad.hasura-app.io

Note: Files that are deployed to the server are files as they are currently on this console. These files may or may not be saved to your github repo.

On this page, which is your coding sort of programming interface; you can see instructions about how to use the console on slash code, slash home. And we can see that we have several different files here.

(Refer Slide Time: 01:28)

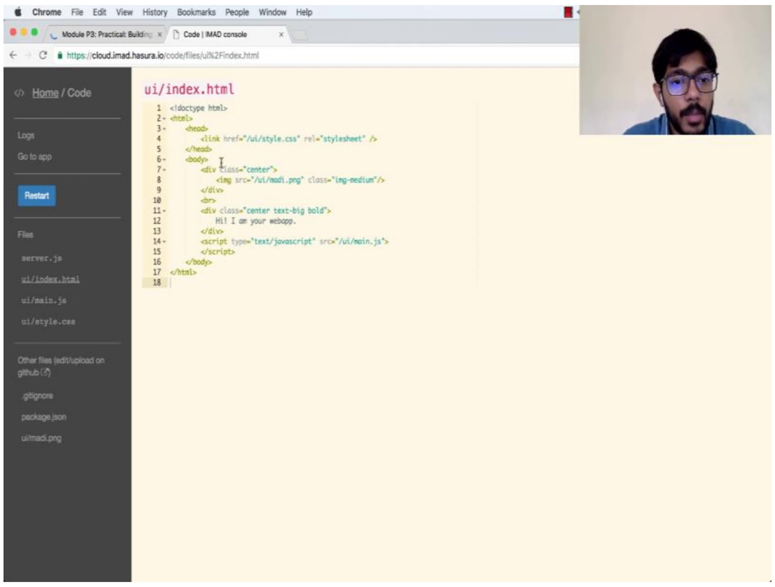


server.js

```
var express = require('express');
var morgan = require('morgan');
var path = require('path');

var app = express();
app.use(morgan('combined'));

app.get('/', function (req, res) {
  res.sendFile(path.join(__dirname, 'ui', 'index.html'));
});

app.get('/ui/style.css', function (req, res) {
  res.sendFile(path.join(__dirname, 'ui', 'style.css'));
});

app.get('/ui/madi.png', function (req, res) {
  res.sendFile(path.join(__dirname, 'ui', 'madi.png'));
});


var port = 8080; // Use 8080 for local development because you might already have apache running on 80
app.listen(8080, function () {
  console.log(`IMAD course app listening on port ${port}!`);
});
```

Let us click on a file, and you can see there is some javascript coded in here.

(Refer Slide Time: 01:32)

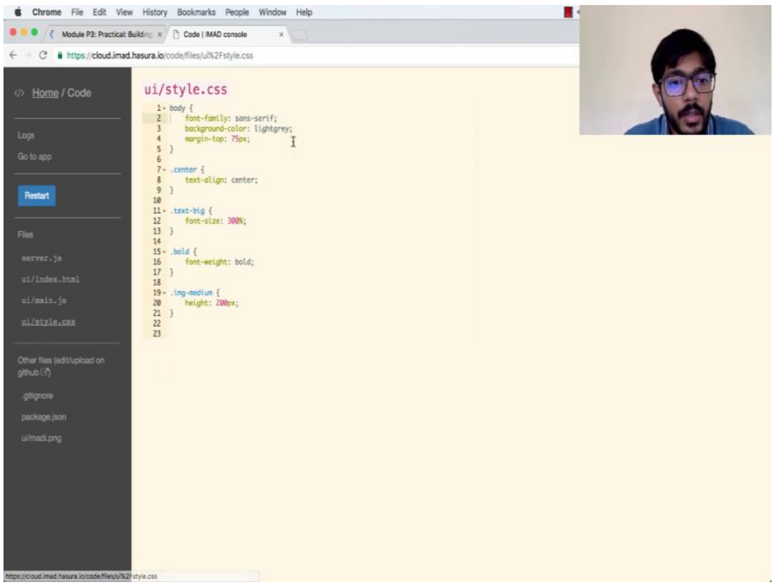


ui/index.html

```
<!doctype html>
<html>
    <head>
        <link href="/ui/style.css" rel="stylesheet" />
    </head>
    <body>
        <div class="center">
            <img src="/ui/madi.png" class="img-medium"/>
        </div>
        <br>
        <div class="center text-big bold">
            Hi! I am your webapp.
        </div>
        <script type="text/javascript" src="/ui/main.js">
        </script>
    </body>
</html>
```

HTML here.

(Refer Slide Time: 01:35)

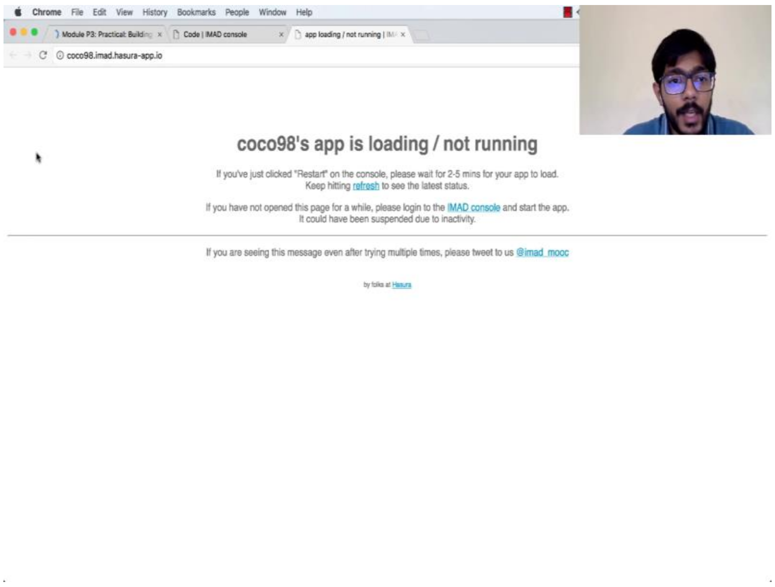

Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Code | IMAD console
https://cloud.imad.hasura.io/code/files/ui%2Fstyle.css
Home / Code
Logs
Go to app
Restart
Files
server.js
ui/index.html
ui/main.js
ui/style.css
Other files (edit/upload on github)
.gitignore
package.json
ui/madi.png

ui/style.css

```css
body {
    font-family: sans-serif;
    background-color: lightgrey;
    margin-top: 75px;
}

.center {
    text-align: center;
}

.text-big {
    font-size: 300%;
}

.bold {
    font-weight: bold;
}

.img-medium {
    height: 200px;
}
```

There is some javascript here and some CSS here. We will understand what these are again in a little bit. For now, let us try to go to our app and see what happens when these source code files are deployed into a webapp.

(Refer Slide Time: 01:51)


Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Code | IMAD console
app loading / not running | IM
coco98.imad.hasura-app.io

coco98's app is loading / not running

If you've just clicked "Restart" on the console, please wait for 2-5 mins for your app to load.
Keep hitting refresh to see the latest status.

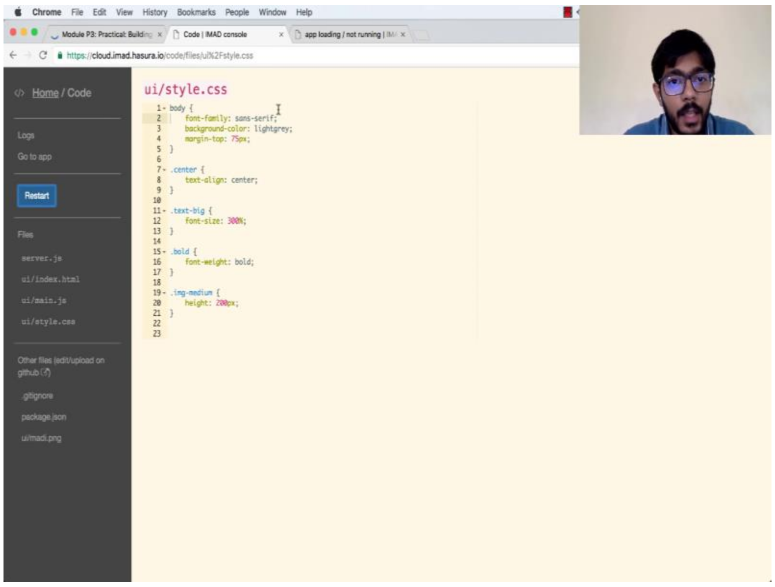
If you have not opened this page for a while, please login to the IMAD console and start the app.
It could have been suspended due to inactivity.

If you are seeing this message even after trying multiple times, please tweet to us @imad_mooc

by folks at Hasura

When I click on it, I get a message saying that my app is not running.

(Refer Slide Time: 01:56)

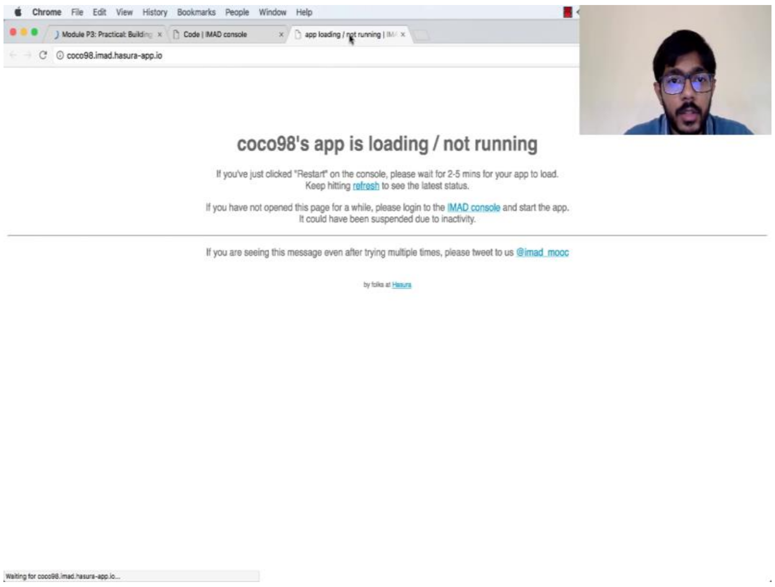


ui/style.css

```
body {
    font-family: sans-serif;
    background-color: lightgrey;
    margin-top: 75px;
}

.center {
    text-align: center;
}

.text-big {
    font-size: 300%;
}

.bold {
    font-weight: bold;
}

.img-medium {
    height: 200px;
}
```

So, I need to restart my app.

(Refer Slide Time: 01:03)



coco98's app is loading / not running

If you've just clicked "Restart" on the console, please wait for 2-5 mins for your app to load.
Keep hitting refresh to see the latest status.

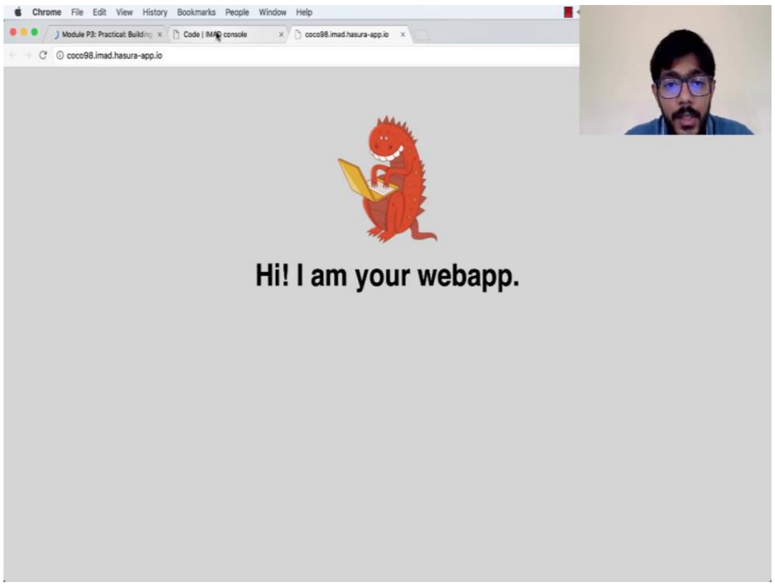
If you have not opened this page for a while, please login to the IMAD console and start the app.
It could have been suspended due to inactivity.

If you are seeing this message even after trying multiple times, please tweet to us @imad_mooc

by folks at Hasura

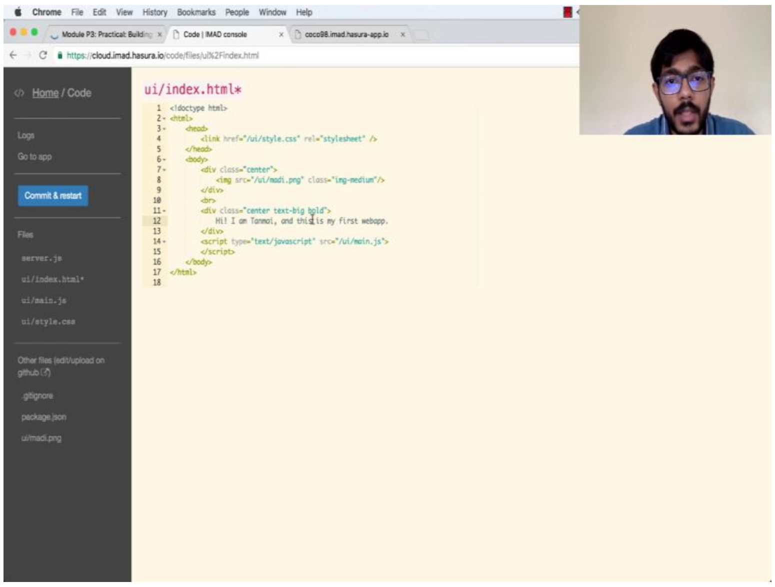
Now, once I click on restart, I can go and try to refresh this page, but the server might take a while to start and so please to be patient for few minutes to let your server start.

(Refer Slide Time: 02:11)



And you can see now with the server is started. And the first HTML page is loaded at Hi! I am your webapp. This is our project with some basic source code files that is already been deployed for us on a particular domain. Just to see how this works.

(Refer Slide Time: 02:30)

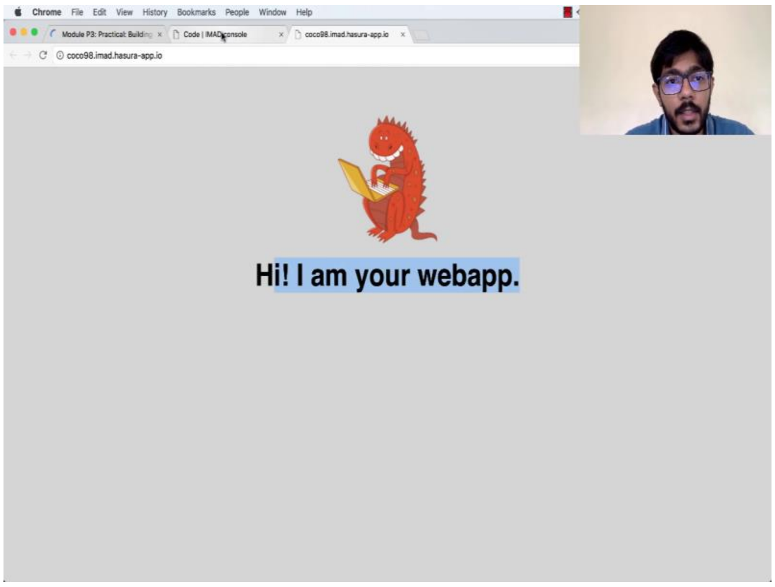


ui/index.html*

```
<!doctype html>
<html>
    <head>
        <link href="/ui/style.css" rel="stylesheet" />
    </head>
    <body>
        <div class="center">
            <img src="/ui/madi.png" class="img-medium"/>
        </div>
        <br>
        <div class="center text-big bold">
            Hi! I am Tanmai, and this is my first webapp.
        </div>
        <script type="text/javascript" src="/ui/main.js">
        </script>
    </body>
</html>
```

Let us go and make some small changes. So, I should change, Hi I am your webapp to Hi I am Tanmai and this is my first webapp. So, I made the changes to the HTML file.

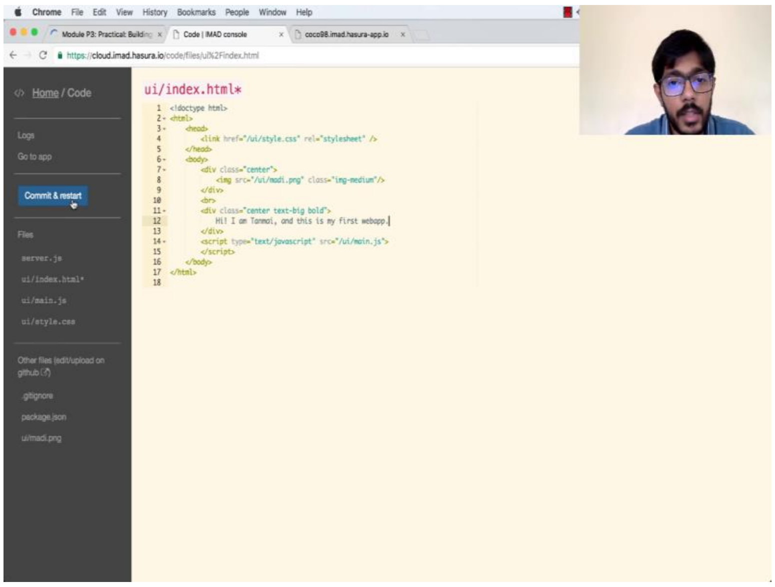
(Refer Slide Time: 02:50)


Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Code | IMAD console
coco98.imad.hasura-app.io
coco98.imad.hasura-app.io

Hi! I am your webapp.

And so hopefully I should be seeing that text replace this text here, this is I am your webapp should get replaced.

(Refer Slide Time: 02:56)

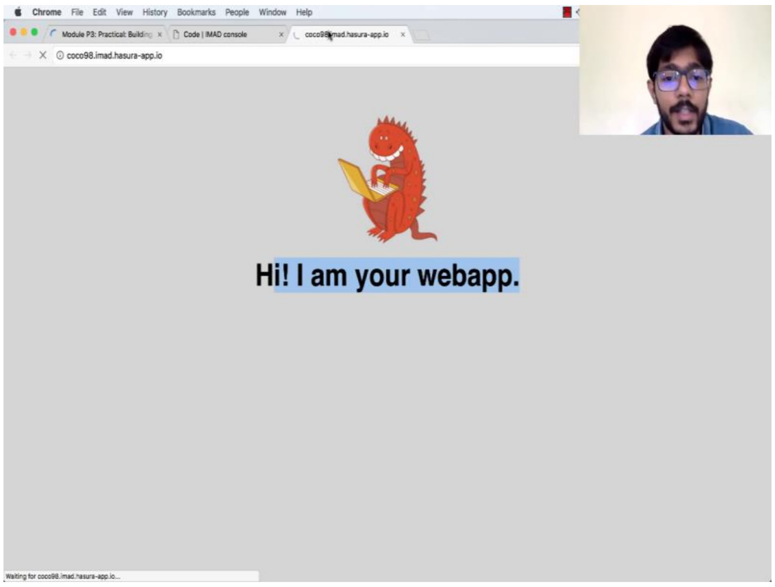

Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Code | IMAD console
coco98.imad.hasura-app.io
https://cloud.imad.hasura.io/code/files/ui%2Findex.html
Home / Code
Logs
Go to app
Commit & restart
Files
server.js
ui/index.html*
ui/main.js
ui/style.css
Other files (edit/upload on github)
.gitignore
package.json
ui/madi.png

ui/index.html*

```
<!doctype html>
<html>
    <head>
        <link href="/ui/style.css" rel="stylesheet" />
    </head>
    <body>
        <div class="center">
            <img src="/ui/madi.png" class="img-medium"/>
        </div>
        <br>
        <div class="center text-big bold">
            Hi! I am Tanmai, and this is my first webapp.
        </div>
        <script type="text/javascript" src="/ui/main.js">
        </script>
    </body>
</html>
```

Let us save this and restart our server. Just go back.

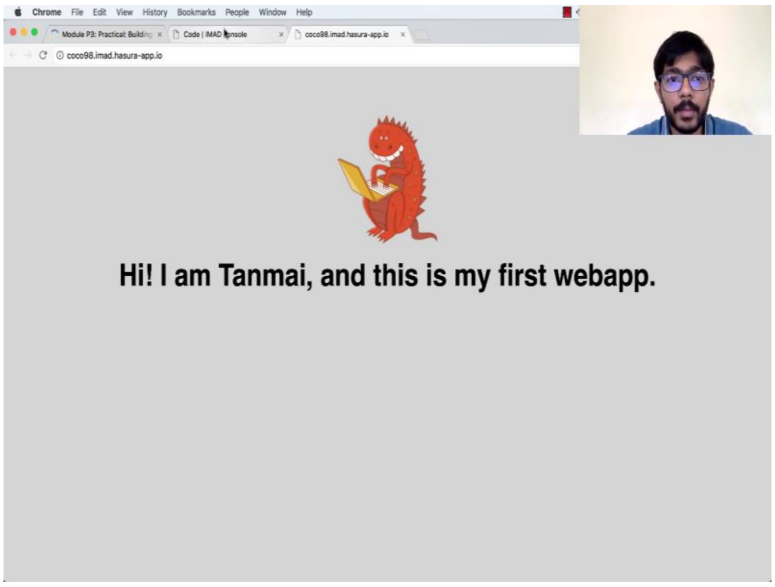
(Refer Slide Time: 03:07)


Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Code | IMAD console
coco98.imad.hasura-app.io
coco98.imad.hasura-app.io
Waiting for coco98.imad.hasura-app.io...

Hi! I am your webapp.

And refresh this please be aware that you will have to give some time, because your files are being copied again your application is being built and is been deployed on the server.

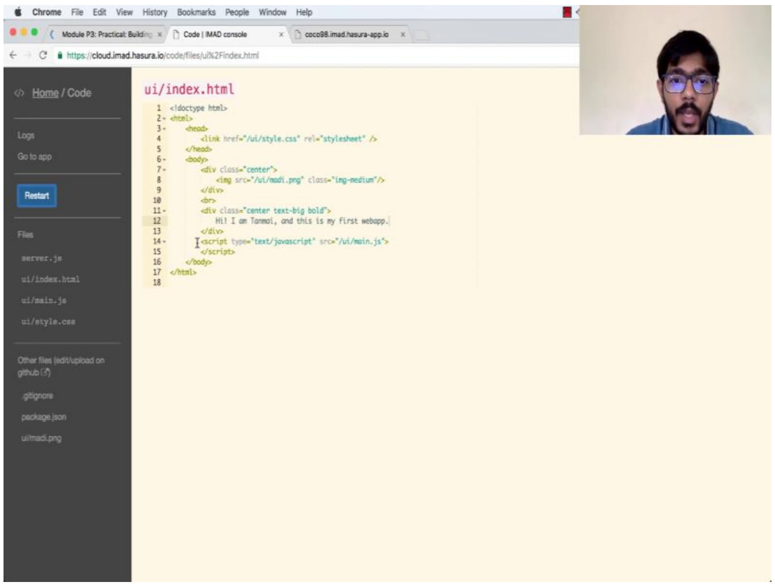
(Refer Slide Time: 03:27)


Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Code | IMAD console
coco98.imad.hasura-app.io
coco98.imad.hasura-app.io

Hi! I am Tanmai, and this is my first webapp.

So, we can refresh to three times, and you can see that the text is changed, and this is how I can change and deploy code on my webapp.

(Refer Slide Time: 03:35)

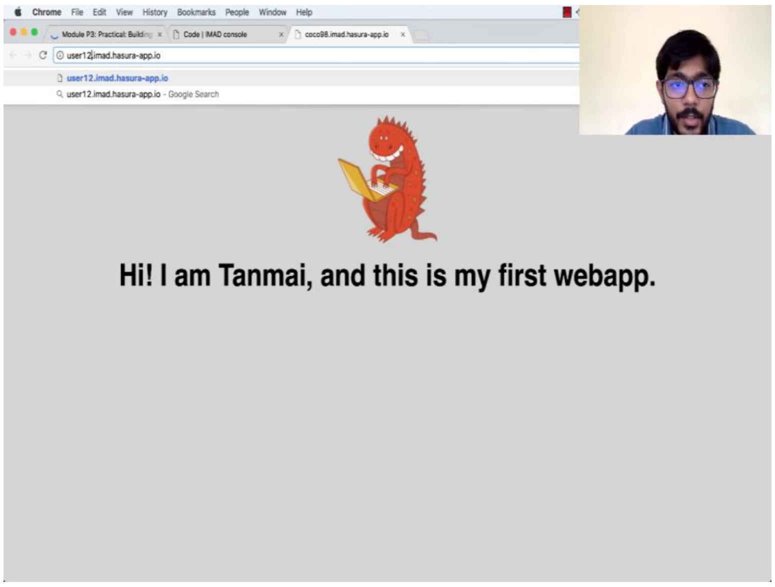

Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Code | IMAD console
coco98.imad.hasura-app.io
https://cloud.imad.hasura.io/code/files/ui%2Findex.html
Home / Code
Logs
Go to app
Restart
Files
server.js
ui/index.html
ui/main.js
ui/style.css
Other files (edit/upload on github)
.gitignore
package.json
ui/madi.png

ui/index.html

```
<!doctype html>
<html>
    <head>
        <link href="/ui/style.css" rel="stylesheet" />
    </head>
    <body>
        <div class="center">
            <img src="/ui/madi.png" class="img-medium"/>
        </div>
        <br>
        <div class="center text-big bold">
            Hi! I am Tanmai, and this is my first webapp.
        </div>
        <script type="text/javascript" src="/ui/main.js">
        </script>
    </body>
</html>
```

Now, let us try to understand how this works what just happened, when we had a bunch of source code files I made some changes I clicked on a button.

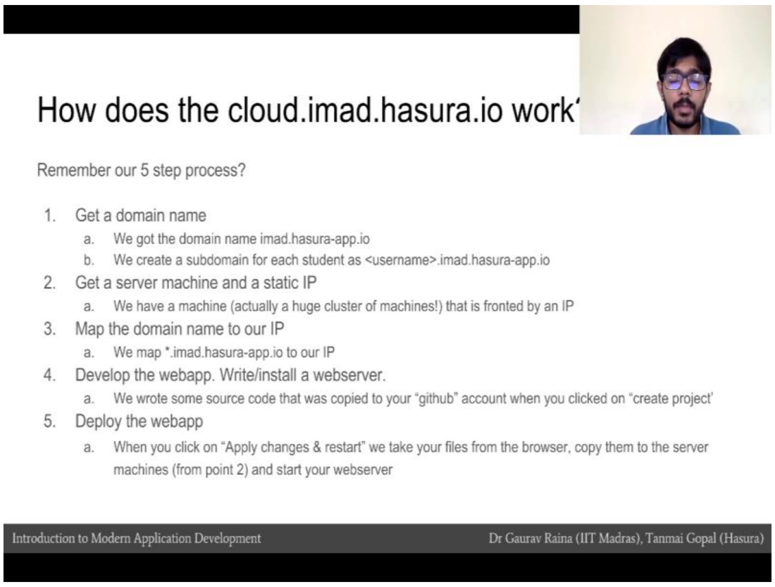
(Refer Slide Time: 03:45)


Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Code | IMAD console
coco98.imad.hasura-app.io
user12.imad.hasura-app.io
user12.imad.hasura-app.io
user12.imad.hasura-app.io - Google Search

Hi! I am Tanmai, and this is my first webapp.

And suddenly, I am able to see my server live on my own domain. For example, if your user name is user 1 2, your app would be on user12.imad.hasura.io, and mine because my user name is coco98, my app is coco98.imad.hasura.io.

(Refer Slide Time: 04:01)



How does the cloud.imad.hasura.io work'

Remember our 5 step process?

1. Get a domain name
   a. We got the domain name imad.hasura-app.io
   b. We create a subdomain for each student as <username>.imad.hasura-app.io
2. Get a server machine and a static IP
   a. We have a machine (actually a huge cluster of machines!) that is fronted by an IP
3. Map the domain name to our IP
   a. We map *.imad.hasura-app.io to our IP
4. Develop the webapp. Write/install a webserver.
   a. We wrote some source code that was copied to your "github" account when you clicked on "create project'
5. Deploy the webapp
   a. When you click on "Apply changes & restart" we take your files from the browser, copy them to the server machines (from point 2) and start your webserver

Introduction to Modern Application Development Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

So, how is this happening, if you remember we had discussed a five-step process last week about how our webapp would be deployed. And we said that the process roughly consist of getting a domain name, getting hosting, linking the domain app to the IP that the hosting provides us with developing the source code or installing a webapp or a web server and then deploying that on that machine, so that that web server is accessible on that IP and on that domain name.

Let us try to revisit that five-step process and understand how this console is helping us. The first thing that we had to do is to get a domain name; in this case, we have already created domain names using sub domains. The domain is hasuraapp.io, on which we have the sub domains username.imad.hasuraapp.io to help us deploy our web apps on these specific domains.

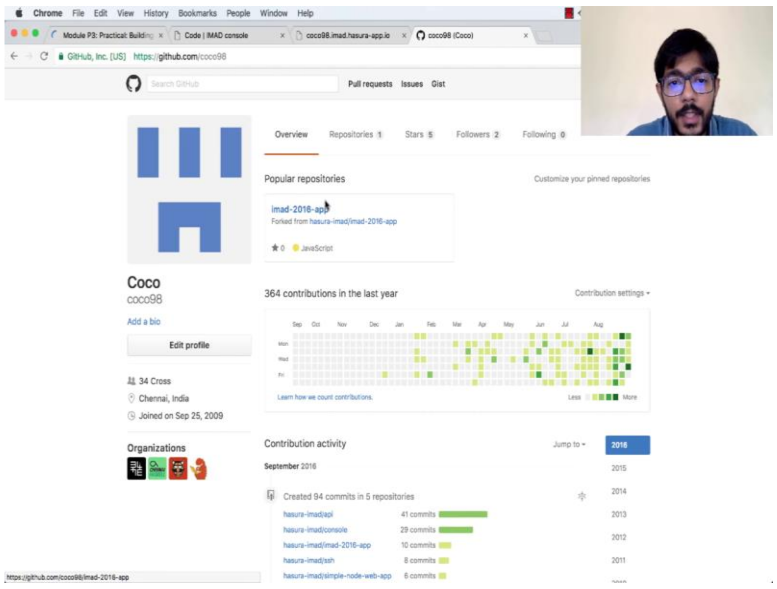
Our second step was to get a server machine in a static IP. In our case, we have got in we have got a machine on actually a large cluster of machines for all the students of this course. And all of them are fronted by a single IP. This is equivalent to a set up we have one large machine which has one IP, and all the servers for all the students will be deployed on this single machine. And then I think that that we did was map the domain to our IP, which we done. All the sub domains at imad.hasuraapp.io, which is to say star.imad.hasuraapp.io, are all domains that are pointing to our IP.

No matter what user name you enter with imad.hasuraapp.io, it will point to the same IP. And you can try to resolve for this by using ping and doing a ping on your user name. imad.hasuraapp.io and seeing where the response of that ping is or what the resolution of that ping is.

The fourth thing that we had to do is develop the webapp or write an install a web server. In our case since, we copied a project a pre existing project because we did not want to write all the code from scratch just to demonstrate this process. We copy the source code files for a webapp. And when we clicked on create project, actually what was happening was that was that project files have been copied from a templated project into our project and that is kind of where the source code files came from.

The fifth step was to deploy the source code files onto our server, so that it is alive at the domain name that we have bought. In this case or in the consoles case, when you click on commit and restart and the blue button mark commit in restart these files are saved to github and then your server is restarted with the latest changes from your github project. Let us now go into little more detail and actually understand what are github project is and what these source code files are.

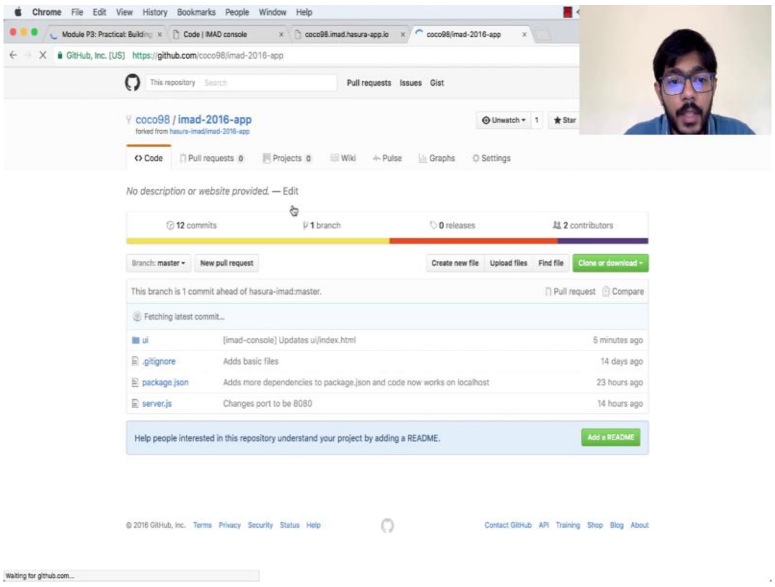
(Refer Slide Time: 06:53)



https://github.com/coco98
Pull requests
Issues
Gist
Overview
Repositories 1
Stars 5
Followers 2
Following 0
Popular repositories
imad-2016-app
Forked from hasura-imad/imad-2016-app
JavaScript
364 contributions in the last year
Coco
coco98
Edit profile
34 Cross
Chennai, India
Joined on Sep 25, 2009
Organizations
Contribution activity
September 2016
Created 94 commits in 5 repositories
hasura-imad/api 41 commits
hasura-imad/console 29 commits
hasura-imad/imad-2016-app 10 commits
hasura-imad/ssh 8 commits
hasura-imad/simple-node-web-app 6 commits

So, if I go to my github account slash coco 98 can see that there is a new project has been created called imad 2016 app. We did not create this project; this project got created automatically when we clicked on the button create project.

(Refer Slide Time: 07:03)


coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

No description or website provided. — Edit

12 commits | 1 branch | 0 releases | 2 contributors

This branch is 1 commit ahead of hasura-imad:master.

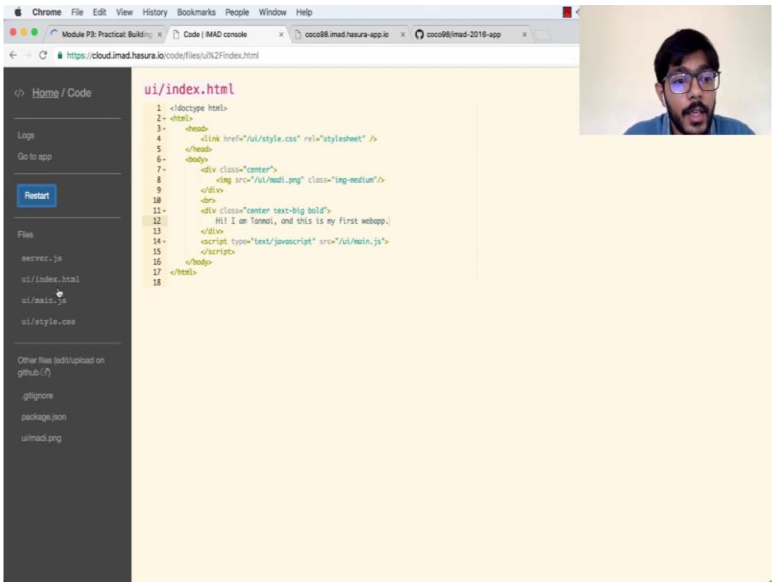
Fetching latest commit…

ui	[imad-console] Updates ui/index.html	5 minutes ago
.gitignore	Adds basic files	14 days ago
package.json	Adds more dependencies to package.json and code now works on localhost	23 hours ago
server.js	Changes port to be 8080	14 hours ago

Help people interested in this repository understand your project by adding a README.

Let us go to this project. And when you go to this project, you can see there are a few files that are already here.

(Refer Slide Time: 07:11)

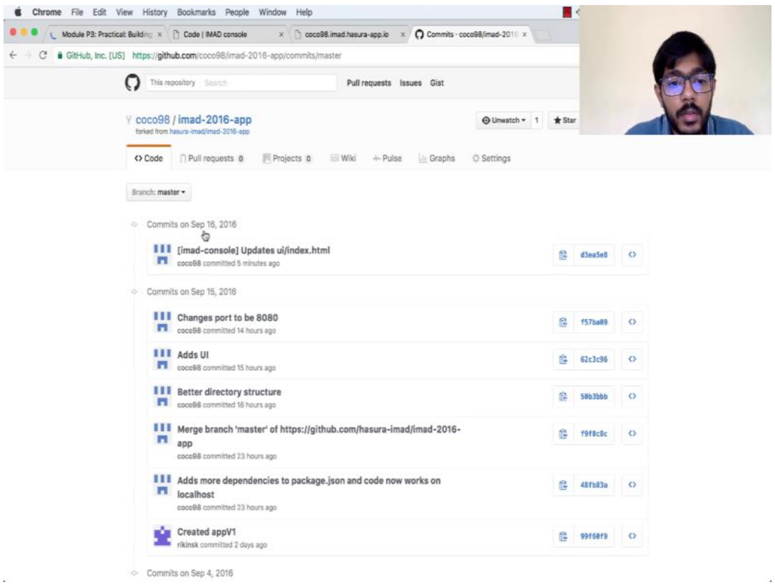


ui/index.html

```html
<!doctype html>
<html>
    <head>
        <link href="/ui/style.css" rel="stylesheet" />
    </head>
    <body>
        <div class="center">
            <img src="/ui/madi.png" class="img-medium"/>
        </div>
        <br>
        <div class="center text-big bold">
            Hi! I am Tanmai, and this is my first webapp.
        </div>
        <script type="text/javascript" src="/ui/main.js">
        </script>
    </body>
</html>
```

And these are the files these are the same files that we are looking at in an IMAD console. The editable files are files which are javascript HTML or CSS files other files are not editable on the IMAD console.

(Refer Slide Time: 07:27)



coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

Branch: master

Commits on Sep 16, 2016

- [imad-console] Updates ui/index.html — coco98 committed 5 minutes ago — d3ea5e8

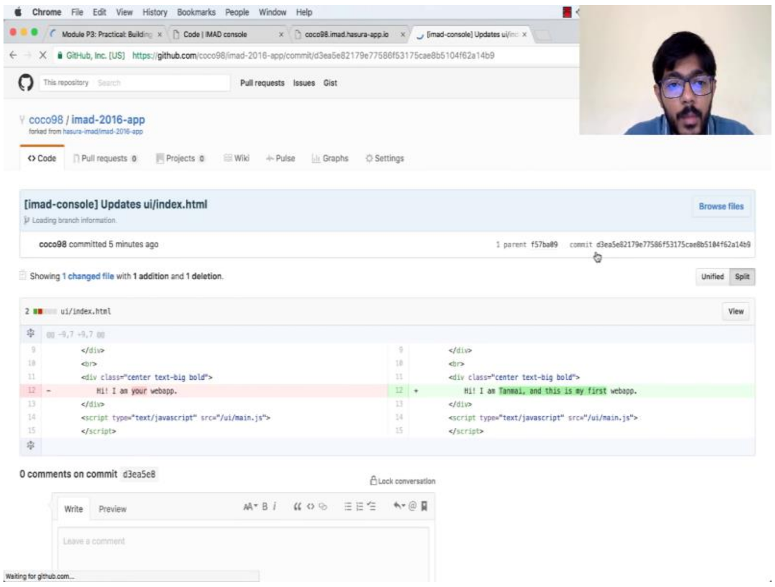
Commits on Sep 15, 2016

- Changes port to be 8080 — coco98 committed 14 hours ago — f57ba09
- Adds UI — coco98 committed 15 hours ago — 62c3c96
- Better directory structure — coco98 committed 18 hours ago — 58b3bbb
- Merge branch 'master' of https://github.com/hasura-imad/imad-2016-app — coco98 committed 23 hours ago — f9f8c8c
- Adds more dependencies to package.json and code now works on localhost — coco98 committed 23 hours ago — 48fb83a
- Created appV1 — rikinsk committed 2 days ago — 99f50f9

Commits on Sep 4, 2016

If you look at the comment history of the project which tells us the activity of the project, we can see that there was a change made from IMAD console, which says update your index dot html, and this was the change that we made. To understand what this change was you can go and click on the commit id.

(Refer Slide Time: 07:41)



[imad-console] Updates ui/index.html

coco98 committed 5 minutes ago — 1 parent f57ba09 — commit d3ea5e82179e77586f53175cae8b5104f62a14b9

Showing 1 changed file with 1 addition and 1 deletion.

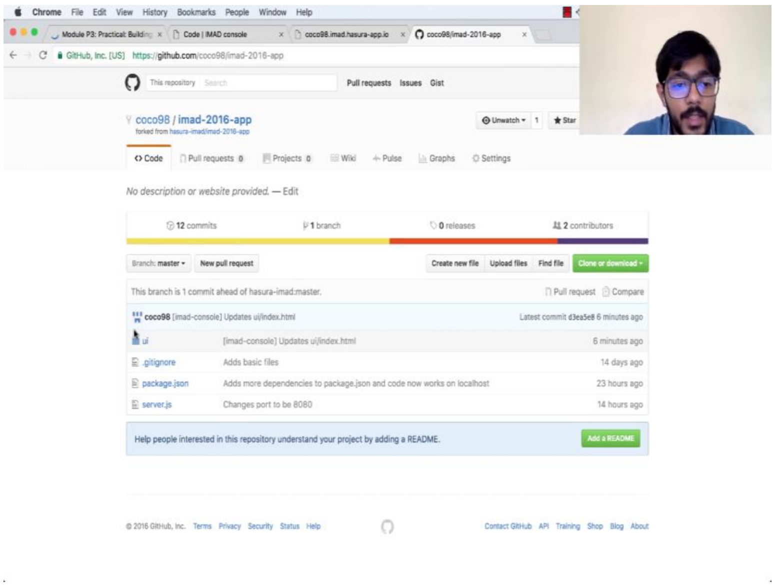
2 ui/index.html

```
@@ -9,7 +9,7 @@
         </div>
         <br>
         <div class="center text-big bold">
-            Hi! I am your webapp.
+            Hi! I am Tanmai, and this is my first webapp.
         </div>
         <script type="text/javascript" src="/ui/main.js">
         </script>
```

0 comments on commit d3ea5e8

And we can see that this was the last change that we made which was hi I am your webapp changed to – Hi! I am Tanmai and this is my first webapp. So, you can see the changes here.

(Refer Slide Time: 07:51)



https://github.com/coco98/imad-2016-app

coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

No description or website provided. — Edit

12 commits | 1 branch | 0 releases | 2 contributors

This branch is 1 commit ahead of hasura-imad:master.

coco98 [imad-console] Updates ui/index.html — Latest commit d3ea5e8 6 minutes ago

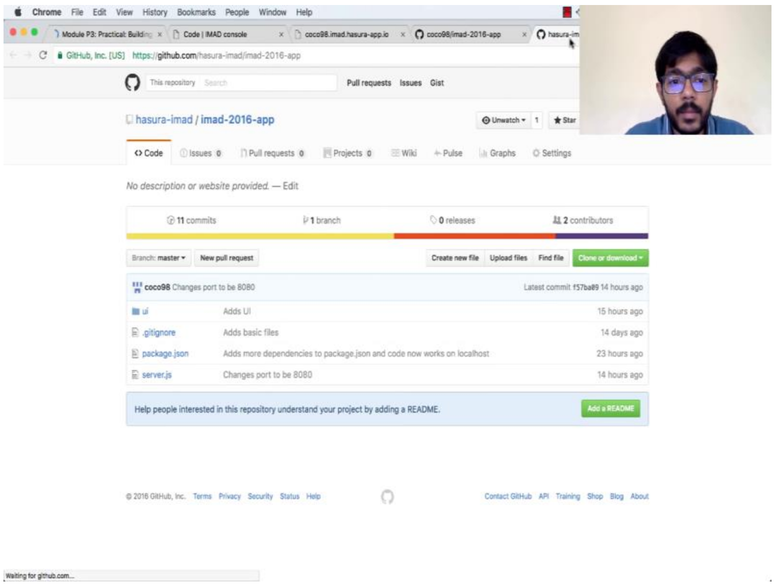
ui	[imad-console] Updates ui/index.html	6 minutes ago
.gitignore	Adds basic files	14 days ago
package.json	Adds more dependencies to package.json and code now works on localhost	23 hours ago
server.js	Changes port to be 8080	14 hours ago

Help people interested in this repository understand your project by adding a README.

© 2016 GitHub, Inc.

Now, where are these files come from, like we said that we been copied from a template project or a base project. And if you go to the title of your own page, you will see that you will see user name slash imad 2016 app forked from hasura imad imad 2016 app.

(Refer Slide Time: 08:08)



https://github.com/hasura-imad/imad-2016-app

hasura-imad / imad-2016-app

No description or website provided. — Edit

11 commits | 1 branch | 0 releases | 2 contributors

coco98 Changes port to be 8080 — Latest commit f57ba89 14 hours ago

ui	Adds UI	15 hours ago
.gitignore	Adds basic files	14 days ago
package.json	Adds more dependencies to package.json and code now works on localhost	23 hours ago
server.js	Changes port to be 8080	14 hours ago

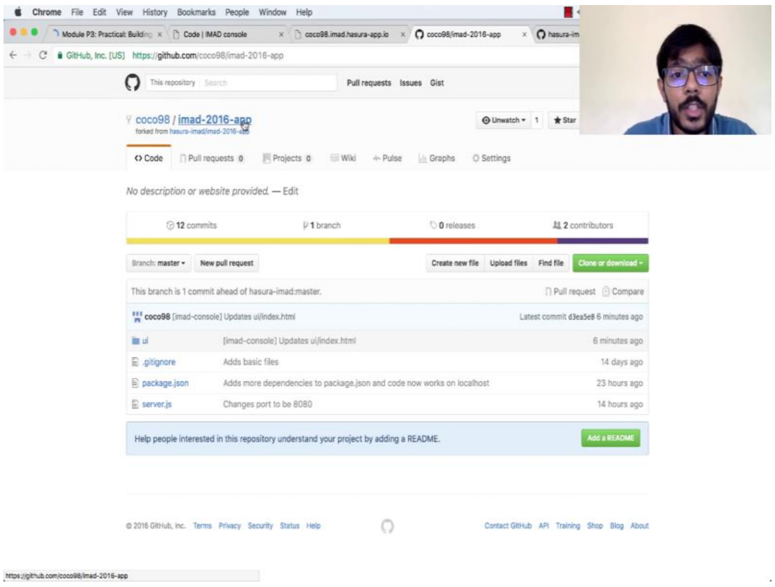
Help people interested in this repository understand your project by adding a README.

© 2016 GitHub, Inc.

Waiting for github.com...

If I click on this link, this is the base project that everybody will take a copy of to create their own project and so start working. And so when I was clicking on create project in the IMAD console the green button, what was happening was this project was been copied into my account.

(Refer Slide Time: 08:23)



coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

No description or website provided. — Edit

12 commits · 1 branch · 0 releases · 2 contributors

This branch is 1 commit ahead of hasura-imad:master.

coco98 [imad-console] Updates ui/index.html — Latest commit d3ea5e8 6 minutes ago

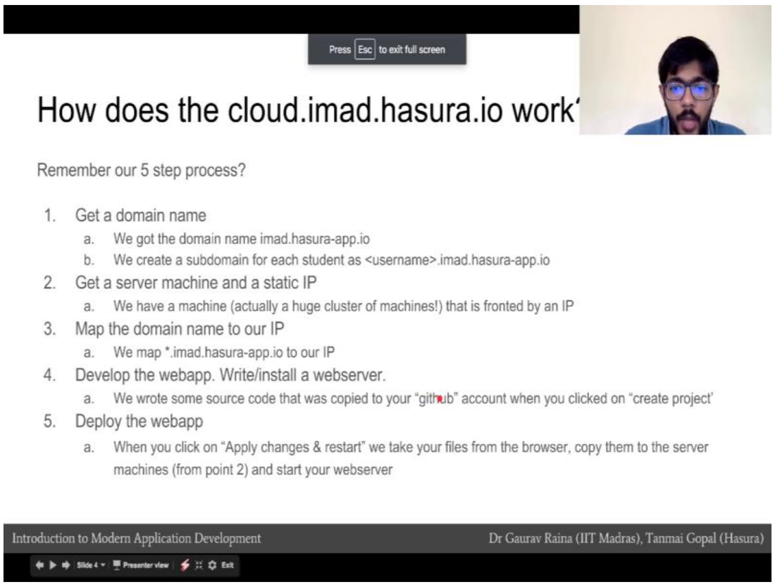
ui	[imad-console] Updates ui/index.html	6 minutes ago
.gitignore	Adds basic files	14 days ago
package.json	Adds more dependencies to package.json and code now works on localhost	23 hours ago
server.js	Changes port to be 8080	14 hours ago

Help people interested in this repository understand your project by adding a README.

© 2016 GitHub, Inc.

And a project was created for me into my account based on this project. And this process of copying from an existing project or copying from an existing git project is called forking.

(Refer Slide Time: 08:36)



How does the cloud.imad.hasura.io work?

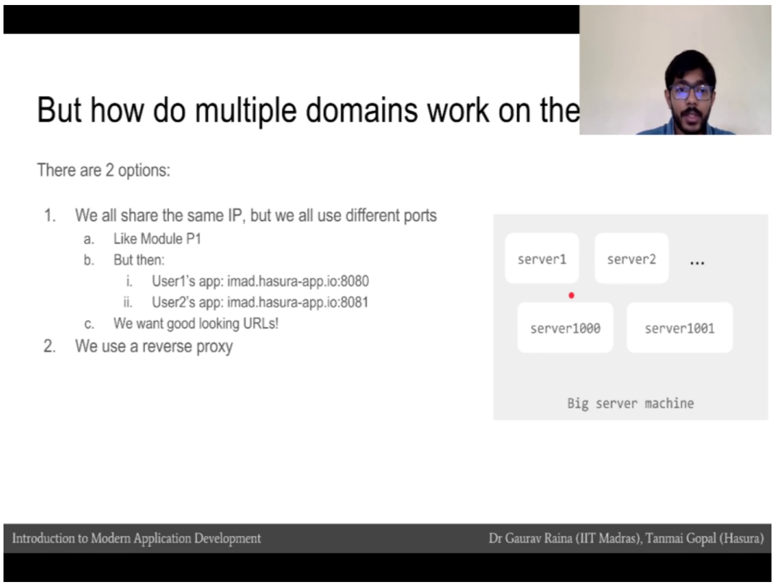
Remember our 5 step process?

1. Get a domain name
   a. We got the domain name imad.hasura-app.io
   b. We create a subdomain for each student as <username>.imad.hasura-app.io
2. Get a server machine and a static IP
   a. We have a machine (actually a huge cluster of machines!) that is fronted by an IP
3. Map the domain name to our IP
   a. We map *.imad.hasura-app.io to our IP
4. Develop the webapp. Write/install a webserver.
   a. We wrote some source code that was copied to your “github” account when you clicked on “create project”
5. Deploy the webapp
   a. When you click on “Apply changes & restart” we take your files from the browser, copy them to the server machines (from point 2) and start your webserver

Introduction to Modern Application Development — Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

Now that we have understood point 4 which is how the source code was copied to our github account and you clicked on create project and surf what is happening in the background. Let us try to understand the fifth step which was how the deployment of the webapp happened. On this console, everybody’s webapp is accessible on user name that imad.hasuraapp.io. And we also said that the all of these domains actually point to the same IP or the same machine.

(Refer Slide Time: 09:02)



But how do multiple domains work on the

There are 2 options:

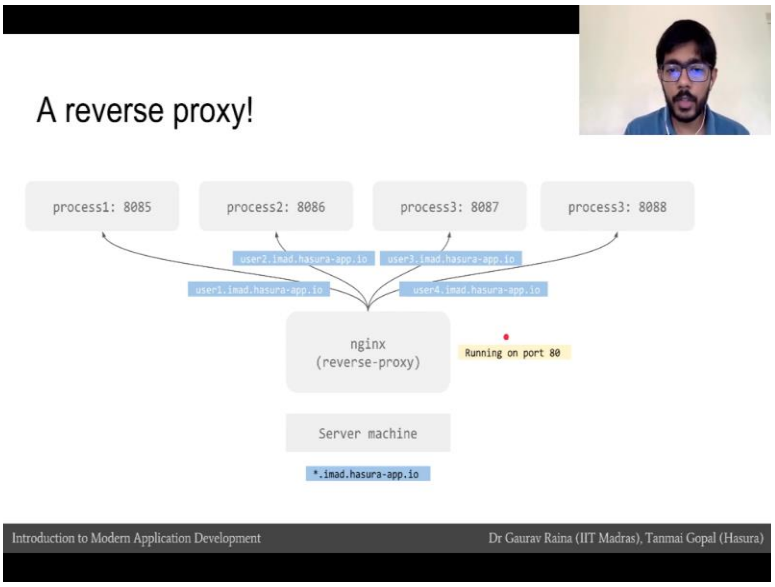
1. We all share the same IP, but we all use different ports
   a. Like Module P1
   b. But then:
      i. User1's app: imad.hasura-app.io:8080
      ii. User2's app: imad.hasura-app.io:8081
   c. We want good looking URLs!
2. We use a reverse proxy

Introduction to Modern Application Development   Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

So, how was this happening, how are multiple domains working on the same IP. There are two ways to achieve. The first way to achieve this is what we covered in the last practical module what we covered in earlier practical module, when we used two web servers on the same computer and we ran them on different ports.

The option here is that we can have a big server machine, and every student server can run on this machine at different ports. And so user one server would be imad.hasuraapp.8080; user 2 would be imad.hasuraapp.io 8081. And you could do this, but this does not give us good looking URLs, this does not give us a feeling that we have our own domain name. And if you really want to have and somehow we want to achieve user name.imad.hasuraapp.io as our domain and the only way to do that is using a reverse proxy.

(Refer Slide Time: 10:02)



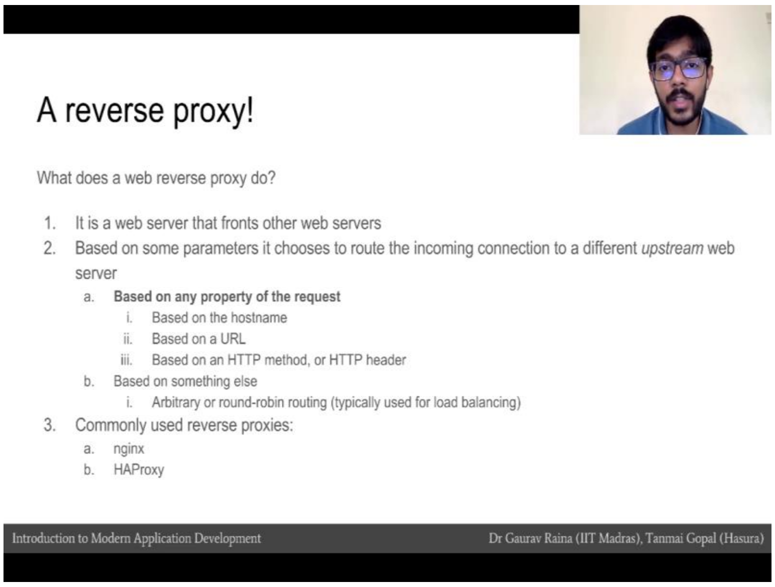
The reverse proxy might be a little complicated to understand and do not worry about it, because it is not extremely important to this course, but it is a nifty concept to understand. Let us look at how the reverse proxy is helping us here. We have star dot imad.hasurapp.io, which is our set of domains pointing to our IP the IP itself is of that of a large server machine. And we somehow need to have multiple web servers running on this single machine.

The first thing that we do is that we set up a web server called nginx, which is the web server listing on port 80. Now as you would remember whenever you connect to a website, the browser by default tries to connect to port 80, you can change that by specifying local host 8080, and so the browser will try to connect to local host at 8080. But if you forget the port 8080, the browser will try to connect two local hosts at the port 80. So, we run a web server an nginx web server on the port 80

But we configure this nginx web server in a special way. What we tells this nginx web server is that if there is an incoming request from let us say for example, userone.imad.hasuraapp.io, and if there is an incoming request that comes to nginx at user one.imad.hasuraapp.io, it forwards that connection to a process that is labelled user one.imad.hasuraapp.io on the same machine. And how that processing running on same machine that process is running on some arbitrary port and we do not care about what port number is running on, because it seems that the response is coming directly from nginx, but it is not.

Nginx is acting as reverse proxy to a process that is sitting what is called upstream to nginx and responding to the request so that means, that userone.imad.hasuraapp.io goes to this process usertwo.imad.hasuraapp.io goes to another process, user three to another process, and user four to another process. This business of creating of front to multiple servers is called reverse proxy. And when we do it for web servers, we call it a web reverse proxy

(Refer Slide Time: 12:15)



A reverse proxy!

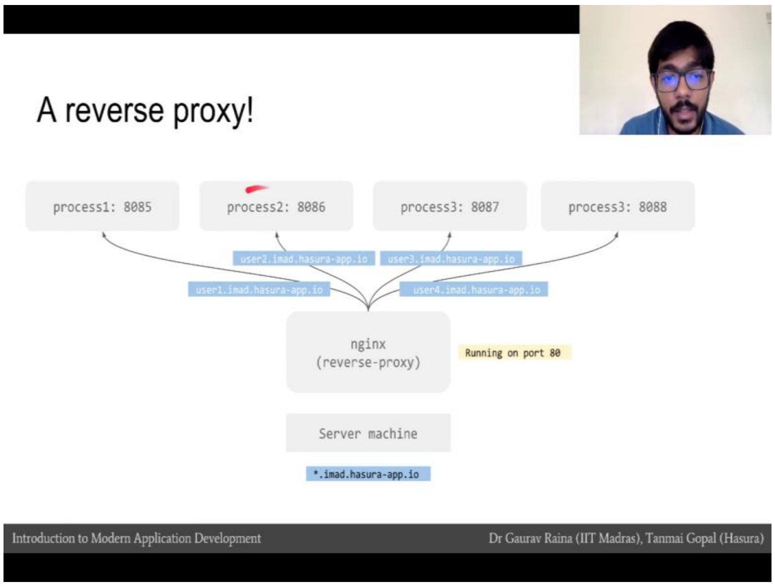
What does a web reverse proxy do?

1. It is a web server that fronts other web servers
2. Based on some parameters it chooses to route the incoming connection to a different *upstream* web server
   a. **Based on any property of the request**
      i. Based on the hostname
      ii. Based on a URL
      iii. Based on an HTTP method, or HTTP header
   b. Based on something else
      i. Arbitrary or round-robin routing (typically used for load balancing)
3. Commonly used reverse proxies:
   a. nginx
   b. HAProxy

Introduction to Modern Application Development Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

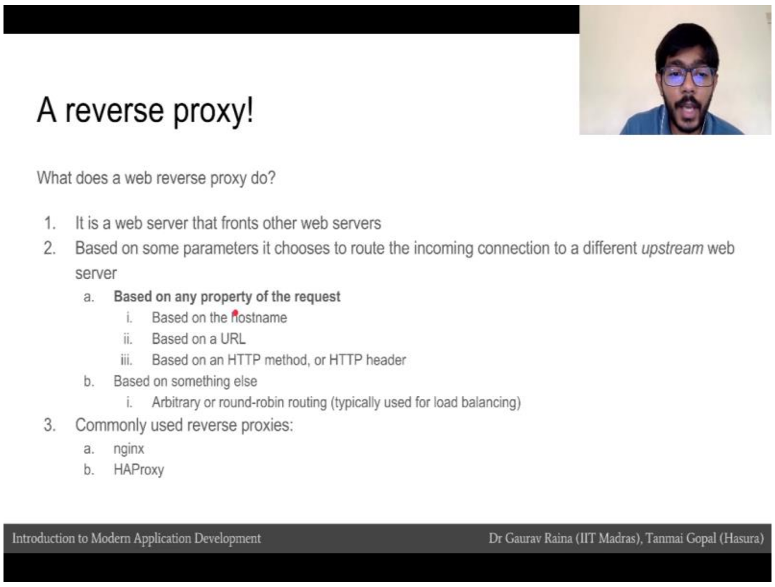
So, what does this reverse proxy do, it is a web server that fronts other web servers which we have understood. And based on certain parameters or based on any property of the incoming request, it can choose to forward it to a particular upstream server, just to revise our definitions again.

(Refer Slide Time: 12:34)



We are calling these processes upstream server and we are calling nginx our web server or the reverse proxy or the web server that is listing on port 80.

(Refer Slide Time: 12:44)



A reverse proxy!

What does a web reverse proxy do?

1. It is a web server that fronts other web servers
2. Based on some parameters it chooses to route the incoming connection to a different *upstream* web server
   a. **Based on any property of the request**
      i. Based on the hostname
      ii. Based on a URL
      iii. Based on an HTTP method, or HTTP header
   b. Based on something else
      i. Arbitrary or round-robin routing (typically used for load balancing)
3. Commonly used reverse proxies:
   a. nginx
   b. HAProxy

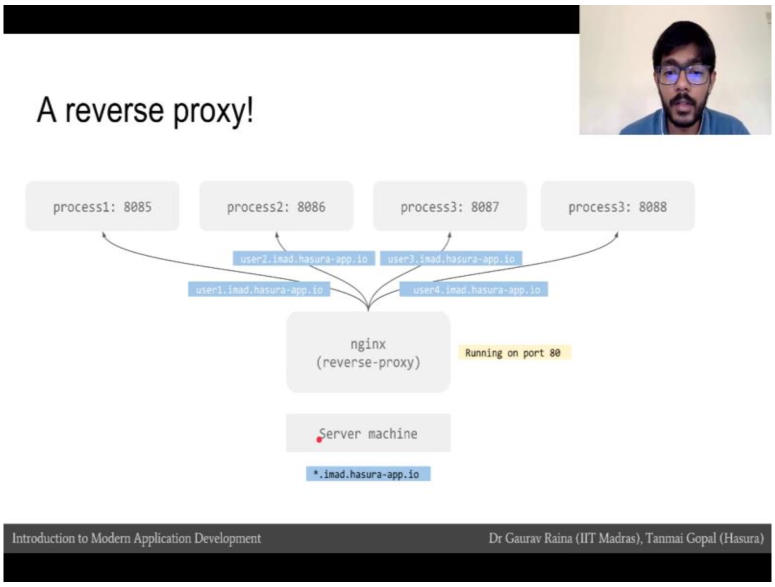
Introduction to Modern Application Development — Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

So, based on any property of the incoming request nginx or the reverse proxy can choose to forward it to a particular upstream server based on that property in our case that property is the host name. So, based on the value of the host name to say userone.imad.hasuraapp.io nginx decides to forward it to a different upstream server but remember that this can actually be done for any property of the request. So, you can create your own reverse proxy that is forwarding to different upstream servers based on the URL path.

For example, imad.hasuraapp.io slash user one will go to the first web server, slash user two will go to the second web server right. In fact, it does not need to be just based on the property of the request object, it could be based on something else all together for example, nginx can decide that it will forward it to upstream server that it will take by random or it will cycle through them in what is called a round-robin method.

It will forward the request once to server 1, then to server 2, then to server 3, then to server 4. This is kind of how you achieve load balancing, because if you have many different web servers and you want to spread your connections or your load your incoming request load across these web servers. Then you have an nginx that sits in front and that forwards the connection to each to a different web server for every new connection. There are two commonly used reverse proxies nginx and HA proxy.

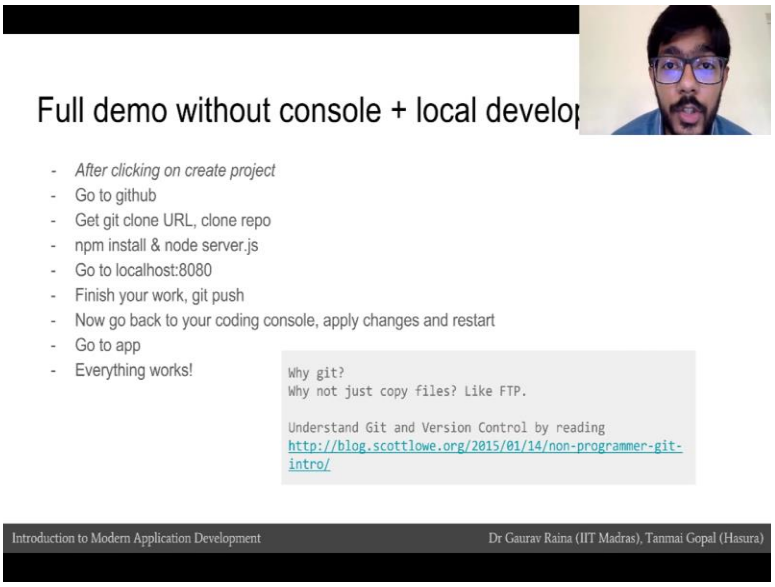
(Refer Slide Time: 14:26)



Incidentally, this is exactly how shared hosting works. When you buy a shared hosting account, the shared hosting provider gives you a large machine; on that machine a several users, and for each user a separate server process is running. But how does everybody manage to get their own servers and all of them seem to be running on port 80, the answer is that they are not running on port 80.

The shared hosting provider has also set up a reverse proxy or a gateway and that gateway based on your own domain name and in this case it is based on sub domain, but in the shared hosting setup it is based on your entire domain name. So, for example, website one dot com could be going to user one, website two dot com could be going to user two, website three dot com could be going to user three. And this is how typical shared hosting setup is done as well.

(Refer Slide Time: 15:10)



Full demo without console + local develo

- *After clicking on create project*
- Go to github
- Get git clone URL, clone repo
- npm install & node server.js
- Go to localhost:8080
- Finish your work, git push
- Now go back to your coding console, apply changes and restart
- Go to app
- Everything works!

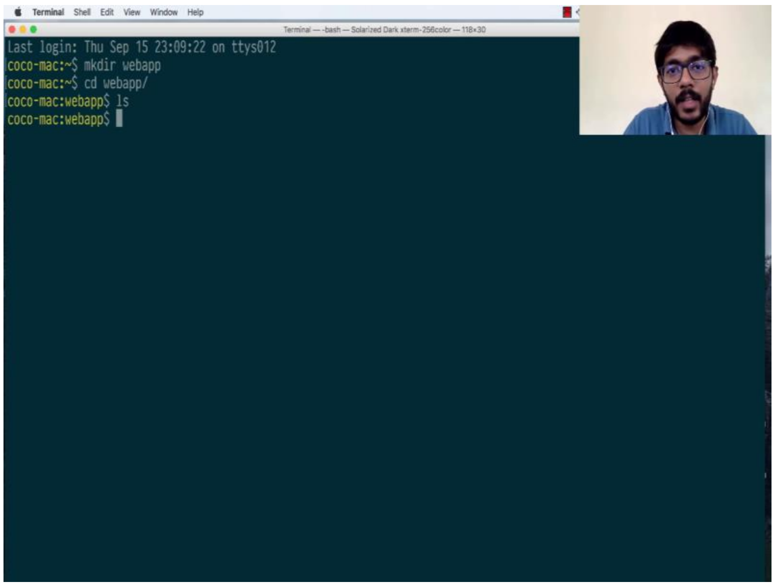
```
Why git?
Why not just copy files? Like FTP.

Understand Git and Version Control by reading
http://blog.scottlowe.org/2015/01/14/non-programmer-git-intro/
```

Introduction to Modern Application Development   Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

Now we have understood what the console, what the IMAD console does, where are source code file are coming from, how are IMAD app is even deployed. And just to sum up what we did on the console was let we created a project. We got a bunch of files. We clicked on a button called restart and suddenly our server was live on username.imad.hasuraapp.io.

Now let us try to do some local development and change the source code of our files on our own computer without needing to connect with the internet or without needing to work directly on IMAD console.

(Refer Slide Time: 15:50)

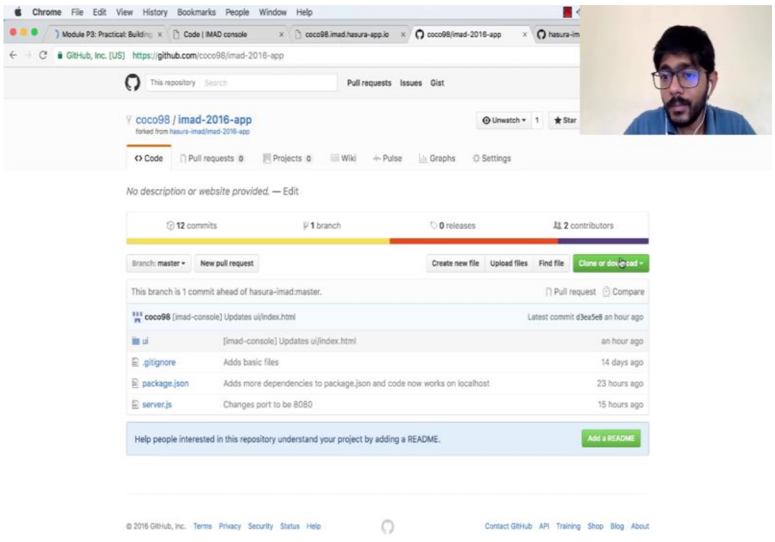

Terminal Shell Edit View Window Help
Terminal — -bash — Solarized Dark xterm-256color — 118×30

```
Last login: Thu Sep 15 23:09:22 on ttys012
coco-mac:~$ mkdir webapp
coco-mac:~$ cd webapp/
coco-mac:webapp$ ls
coco-mac:webapp$ 
```

Let us try to start our local development. The first thing that we will do is open up our terminal. Second, I am going to zoom this up a little bit, so that is clear for everybody. Now I will create a new directly where I want to start doing my work.

So, I will call that nkdir webapp. And I go inside that webapp. Now I want to copy my source code files here, the same source code files that are on my github account, the same source code files that have been taken from my github account putting the IMAD console, I also want them here.

(Refer Slide Time: 16:27)


Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Building
Code | IMAD console
coco98.imad.hasura-app.io
coco98/imad-2016-app
GitHub, Inc. [US] https://github.com/coco98/imad-2016-app
This repository Search
Pull requests Issues Gist
Unwatch 1
Star
Code
Pull requests 0
Projects 0
Wiki
Pulse
Graphs
Settings
Branch: master
New pull request
Create new file
Upload files
Find file
Clone or download
Pull request
Compare
Add a README
© 2016 GitHub, Inc. Terms Privacy Security Status Help
Contact GitHub API Training Shop Blog About

coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

No description or website provided. — Edit

12 commits 1 branch 0 releases 2 contributors

This branch is 1 commit ahead of hasura-imad:master.

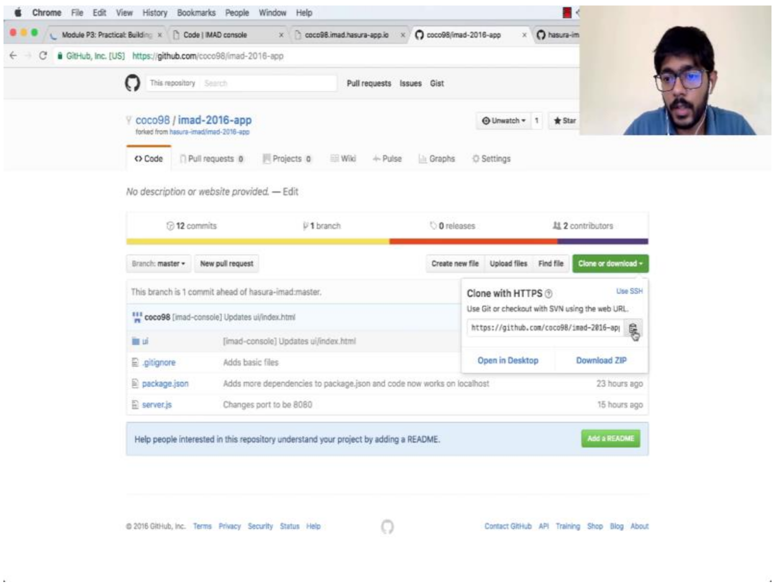
coco98 [imad-console] Updates ui/index.html Latest commit d3ea5e8 an hour ago

ui	[imad-console] Updates ui/index.html	an hour ago
.gitignore	Adds basic files	14 days ago
package.json	Adds more dependencies to package.json and code now works on localhost	23 hours ago
server.js	Changes port to be 8080	15 hours ago

Help people interested in this repository understand your project by adding a README.

So, what I am going to do is go to my github.com project I am going to click on the button called clone or download.

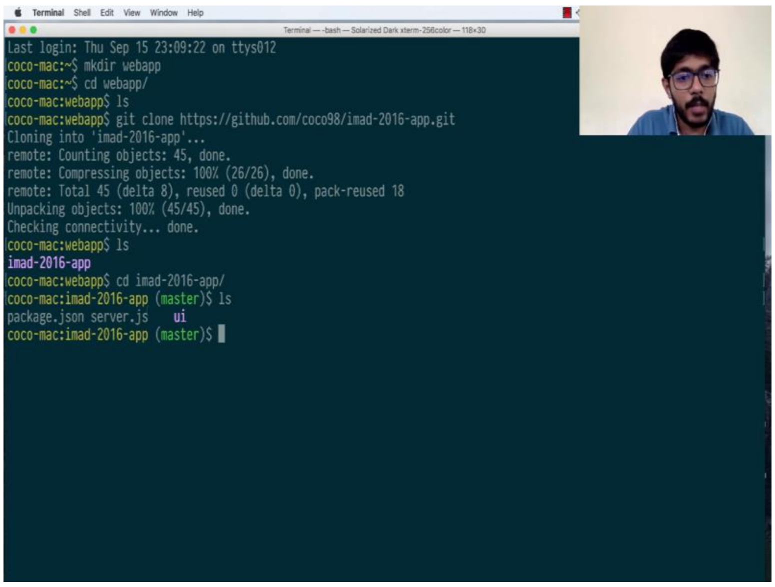
(Refer Slide Time: 16:29)



```
This branch is 1 commit ahead of hasura-imad:master.
coco98 [imad-console] Updates ui/index.html
ui            [imad-console] Updates ui/index.html
.gitignore    Adds basic files
package.json  Adds more dependencies to package.json and code now works on localhost   23 hours ago
server.js     Changes port to be 8080                                                  15 hours ago
Help people interested in this repository understand your project by adding a README.
```

I am going to copy the URL.

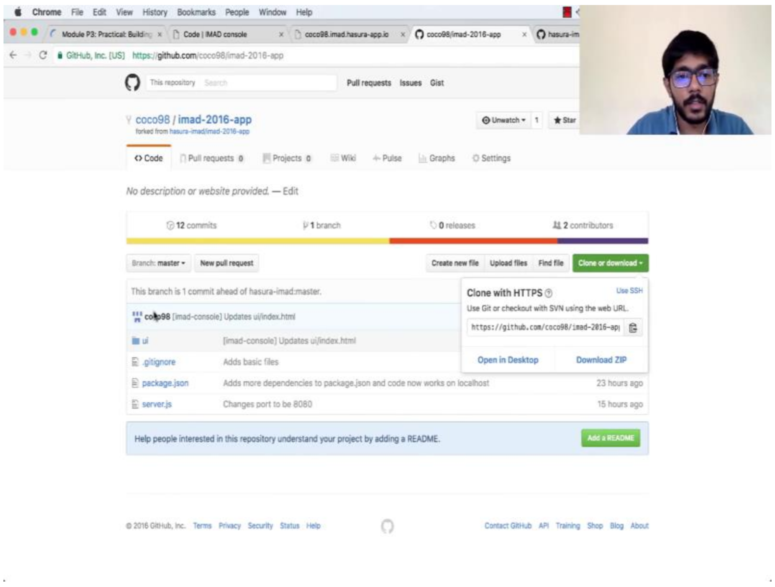
(Refer Slide Time: 16:33)



```
Last login: Thu Sep 15 23:09:22 on ttys012
coco-mac:~$ mkdir webapp
coco-mac:~$ cd webapp/
coco-mac:webapp$ ls
coco-mac:webapp$ git clone https://github.com/coco98/imad-2016-app.git
Cloning into 'imad-2016-app'...
remote: Counting objects: 45, done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 45 (delta 8), reused 0 (delta 0), pack-reused 18
Unpacking objects: 100% (45/45), done.
Checking connectivity... done.
coco-mac:webapp$ ls
imad-2016-app
coco-mac:webapp$ cd imad-2016-app/
coco-mac:imad-2016-app (master)$ ls
package.json server.js    ui
coco-mac:imad-2016-app (master)$
```

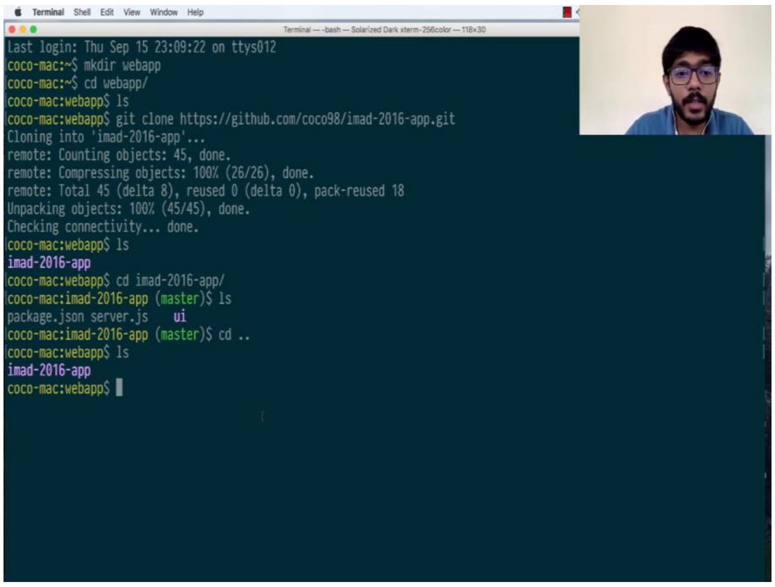
And then I run a command called git clone. And if I do ls to see what files have been created. I can see that new directory has been created called IMAD 2016 app; if I go inside this directory.

(Refer Slide Time: 16:51)


coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app
Code
Pull requests 0
Projects 0
Wiki
Pulse
Graphs
Settings
No description or website provided. — Edit
12 commits
1 branch
0 releases
2 contributors
Branch: master
New pull request
Create new file
Upload files
Find file
Clone or download
This branch is 1 commit ahead of hasura-imad:master.
coco98 [imad-console] Updates ui/index.html
Clone with HTTPS
Use SSH
Use Git or checkout with SVN using the web URL.
https://github.com/coco98/imad-2016-app
Open in Desktop
Download ZIP
ui
[imad-console] Updates ui/index.html
.gitignore
Adds basic files
package.json
Adds more dependencies to package.json and code now works on localhost
23 hours ago
server.js
Changes port to be 8080
15 hours ago
Help people interested in this repository understand your project by adding a README.
Add a README

I can see that those same files are here the files I could present here.

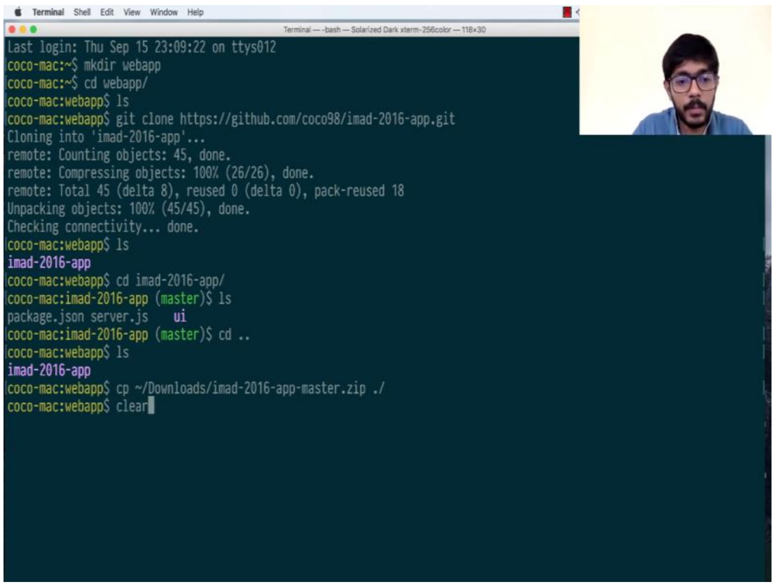
(Refer Slide Time: 16:55)



```
Last login: Thu Sep 15 23:09:22 on ttys012
coco-mac:~$ mkdir webapp
coco-mac:~$ cd webapp/
coco-mac:webapp$ ls
coco-mac:webapp$ git clone https://github.com/coco98/imad-2016-app.git
Cloning into 'imad-2016-app'...
remote: Counting objects: 45, done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 45 (delta 8), reused 0 (delta 0), pack-reused 18
Unpacking objects: 100% (45/45), done.
Checking connectivity... done.
coco-mac:webapp$ ls
imad-2016-app
coco-mac:webapp$ cd imad-2016-app/
coco-mac:imad-2016-app (master)$ ls
package.json server.js    ui
coco-mac:imad-2016-app (master)$ cd ..
coco-mac:webapp$ ls
imad-2016-app
coco-mac:webapp$ 
```

This process was equivalent to taking a downloading a zip file and unzipping the zipped file. In fact, we can do that. Let us try to go download zip you can see that the zip file has been downloaded.

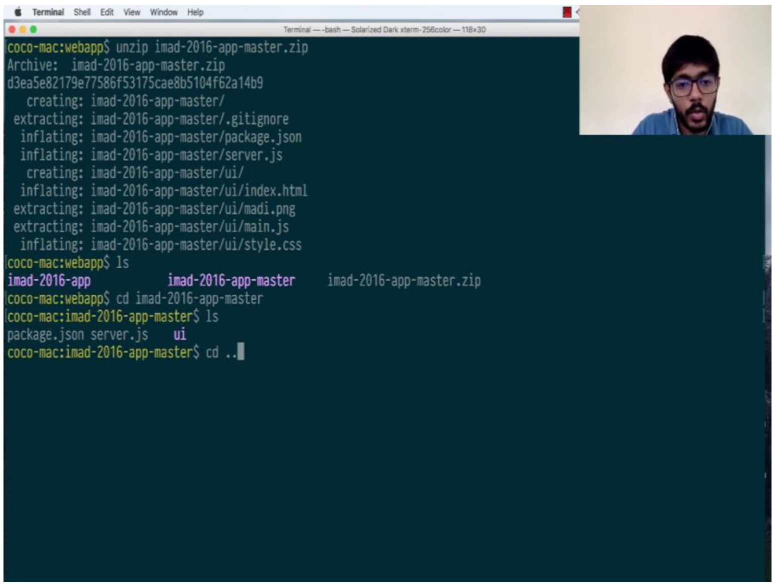
(Refer Slide Time: 17:07)


Terminal Shell Edit View Window Help
Terminal — -bash — Solarized Dark xterm-256color — 118×30

```
Last login: Thu Sep 15 23:09:22 on ttys012
coco-mac:~$ mkdir webapp
coco-mac:~$ cd webapp/
coco-mac:webapp$ ls
coco-mac:webapp$ git clone https://github.com/coco98/imad-2016-app.git
Cloning into 'imad-2016-app'...
remote: Counting objects: 45, done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 45 (delta 8), reused 0 (delta 0), pack-reused 18
Unpacking objects: 100% (45/45), done.
Checking connectivity... done.
coco-mac:webapp$ ls
imad-2016-app
coco-mac:webapp$ cd imad-2016-app/
coco-mac:imad-2016-app (master)$ ls
package.json server.js    ui
coco-mac:imad-2016-app (master)$ cd ..
coco-mac:webapp$ ls
imad-2016-app
coco-mac:webapp$ cp ~/Downloads/imad-2016-app-master.zip ./
coco-mac:webapp$ clear
```

Let me copy the zip file from my download folder, I just clear the screen.

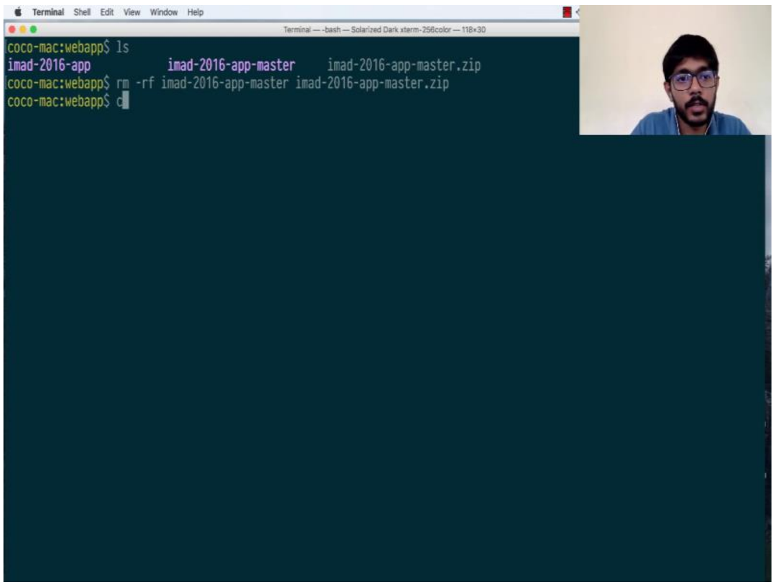
(Refer Slide Time: 17:19)


Terminal Shell Edit View Window Help
Terminal — -bash — Solarized Dark xterm-256color — 118×30

```
coco-mac:webapp$ unzip imad-2016-app-master.zip
Archive:  imad-2016-app-master.zip
d3ea5e82179e77586f53175cae8b5104f62a14b9
   creating: imad-2016-app-master/
 extracting: imad-2016-app-master/.gitignore
  inflating: imad-2016-app-master/package.json
  inflating: imad-2016-app-master/server.js
   creating: imad-2016-app-master/ui/
  inflating: imad-2016-app-master/ui/index.html
 extracting: imad-2016-app-master/ui/madi.png
 extracting: imad-2016-app-master/ui/main.js
  inflating: imad-2016-app-master/ui/style.css
coco-mac:webapp$ ls
imad-2016-app            imad-2016-app-master     imad-2016-app-master.zip
coco-mac:webapp$ cd imad-2016-app-master
coco-mac:imad-2016-app-master$ ls
package.json server.js    ui
coco-mac:imad-2016-app-master$ cd ..
```

Now, I unzip that and you can see that new directory called IMAD 2016 master has come up. And this also contains the same files. So, what I did with git clone was equivalent was similar to downloading a zip file and unzipping it.

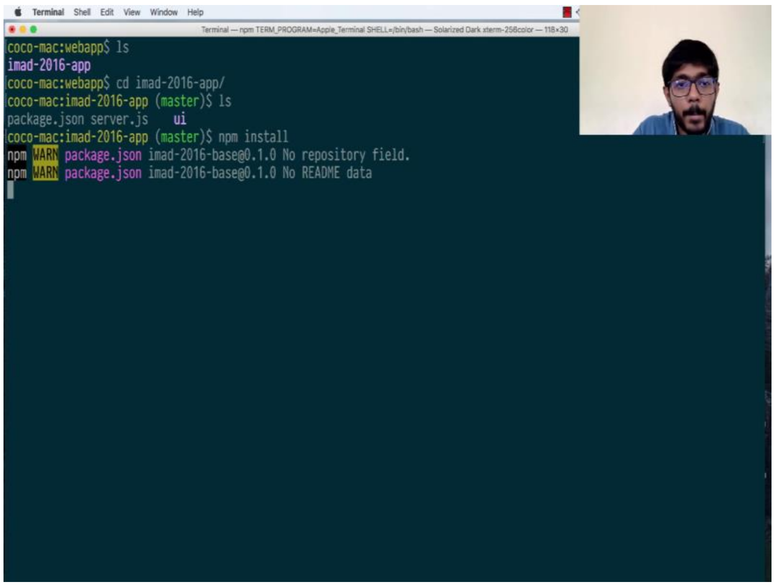
(Refer Slide Time: 17:48)



```
coco-mac:webapp$ ls
imad-2016-app          imad-2016-app-master     imad-2016-app-master.zip
coco-mac:webapp$ rm -rf imad-2016-app-master imad-2016-app-master.zip
coco-mac:webapp$ c
```

Let us remove that the duplicate copies and I will remove the zipped file also.

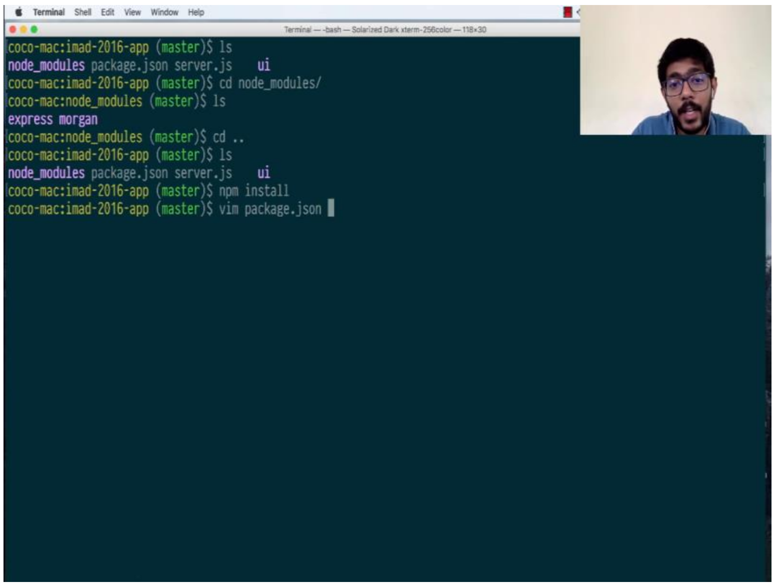
(Refer Slide Time: 17:57)



```
coco-mac:webapp$ ls
imad-2016-app
coco-mac:webapp$ cd imad-2016-app/
coco-mac:imad-2016-app (master)$ ls
package.json server.js    ui
coco-mac:imad-2016-app (master)$ npm install
npm WARN package.json imad-2016-base@0.1.0 No repository field.
npm WARN package.json imad-2016-base@0.1.0 No README data
```

I go inside this folder which was my source code, and now I want to run my server here. I know that this is a nodejs app and so I need to install all the dependencies are required. Once again make sure that you have nodejs and npm installed, and then you can use this to run the command npm install, this will automatically detect what the dependencies of our application are and we are install those dependencies.

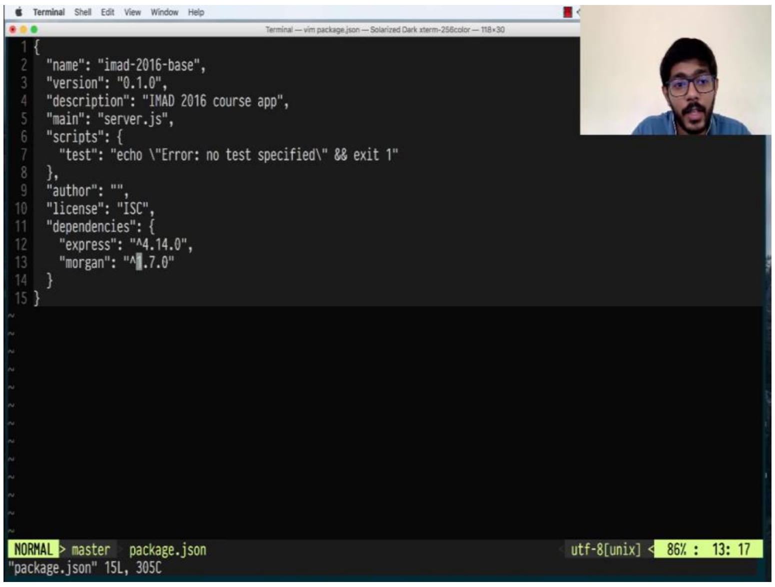
Wait for a second, while it is downloading. And if you list the directories again, you will see that a new folder has been created called node modules. And if you go inside this folder called node modules, you will see that some files have been downloaded these are the dependencies of our file.


Terminal Shell Edit View Window Help
Terminal — -bash — Solarized Dark xterm-256color — 118×30

```
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ cd node_modules/
coco-mac:node_modules (master)$ ls
express morgan
coco-mac:node_modules (master)$ cd ..
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ npm install
coco-mac:imad-2016-app (master)$ vim package.json
```

Remember that there is no magic here right, for example, if you think that n p m install is a magical command that is somehow detecting what the dependencies are and installing them it is because all of these dependencies are in a file called package.json.

(Refer Slide Time: 18:54)

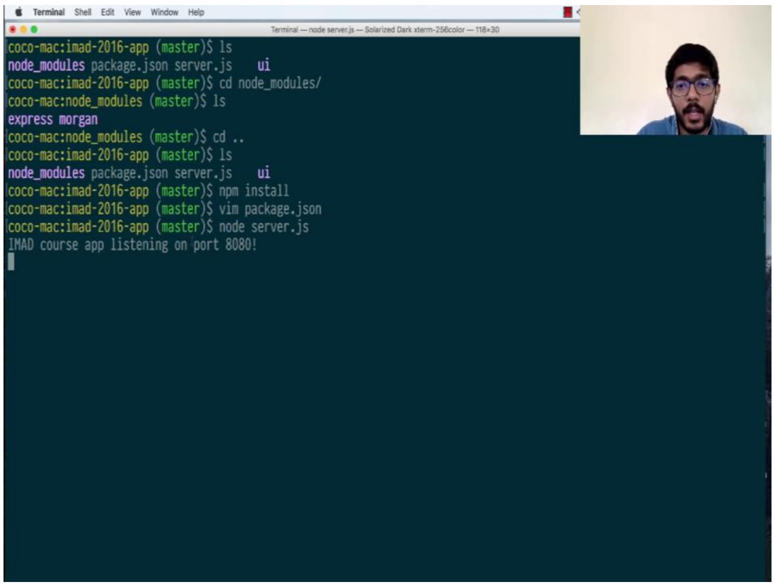

Terminal Shell Edit View Window Help
Terminal — vim package.json — Solarized Dark xterm-256color — 118×30

```
{
  "name": "imad-2016-base",
  "version": "0.1.0",
  "description": "IMAD 2016 course app",
  "main": "server.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.14.0",
    "morgan": "^1.7.0"
  }
}
```

```
NORMAL > master  package.json                                      utf-8[unix] < 86% :  13: 17
"package.json" 15L, 305C
```

And if you look at this and if you look at this there is of there is a json object and this contains the word dependencies and it has two of the dependencies listed out and their (Refer Time: 19:05) numbers.

(Refer Slide Time: 19:08)

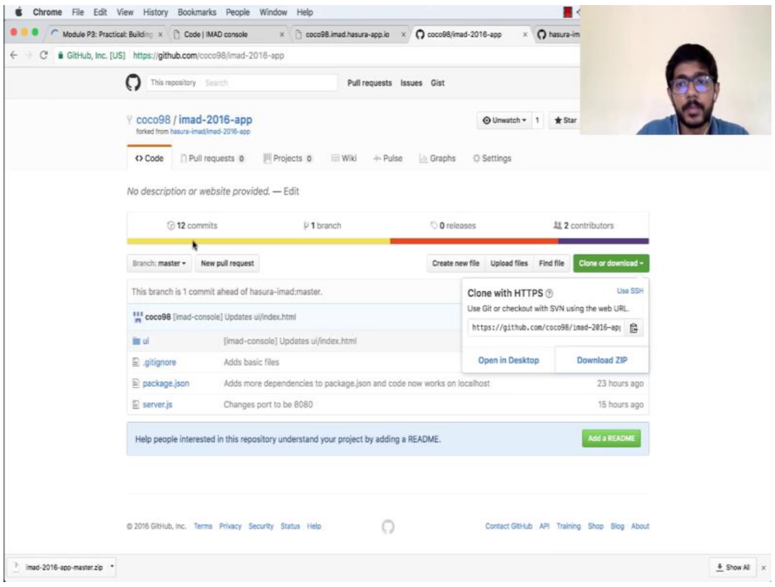


Terminal Shell Edit View Window Help
Terminal — node server.js — Solarized Dark xterm-256color — 118×30

```
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ cd node_modules/
coco-mac:node_modules (master)$ ls
express morgan
coco-mac:node_modules (master)$ cd ..
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ npm install
coco-mac:imad-2016-app (master)$ vim package.json
coco-mac:imad-2016-app (master)$ node server.js
IMAD course app listening on port 8080!
```

That is how npm install knows what modules to install. Now once our module is installed, all we need to do was run node server dot js, and it says that our course app is listing on port 8080.

(Refer Slide Time: 19:23)



coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

No description or website provided. — Edit

12 commits | 1 branch | 0 releases | 2 contributors

This branch is 1 commit ahead of hasura-imad:master.

coco98 [imad-console] Updates ui/index.html

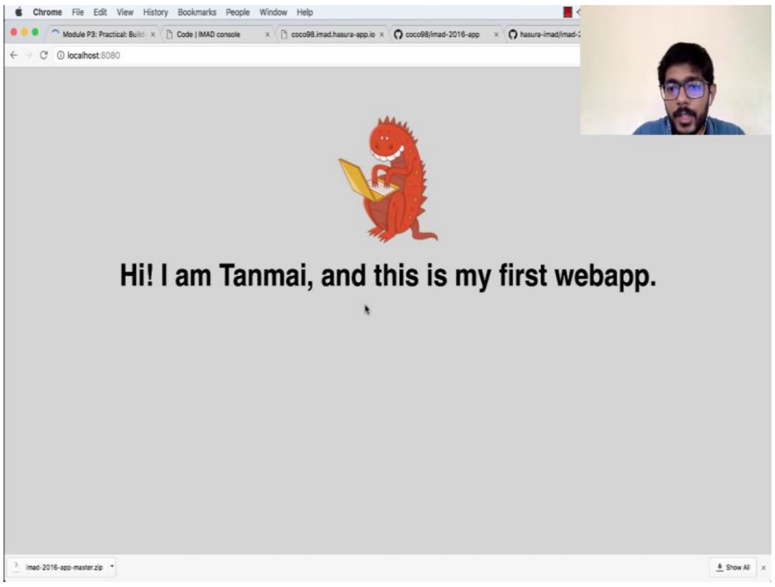
ui	[imad-console] Updates ui/index.html	
.gitignore	Adds basic files	
package.json	Adds more dependencies to package.json and code now works on localhost	23 hours ago
server.js	Changes port to be 8080	15 hours ago

Clone with HTTPS
Use Git or checkout with SVN using the web URL.
https://github.com/coco98/imad-2016-ap

Help people interested in this repository understand your project by adding a README.

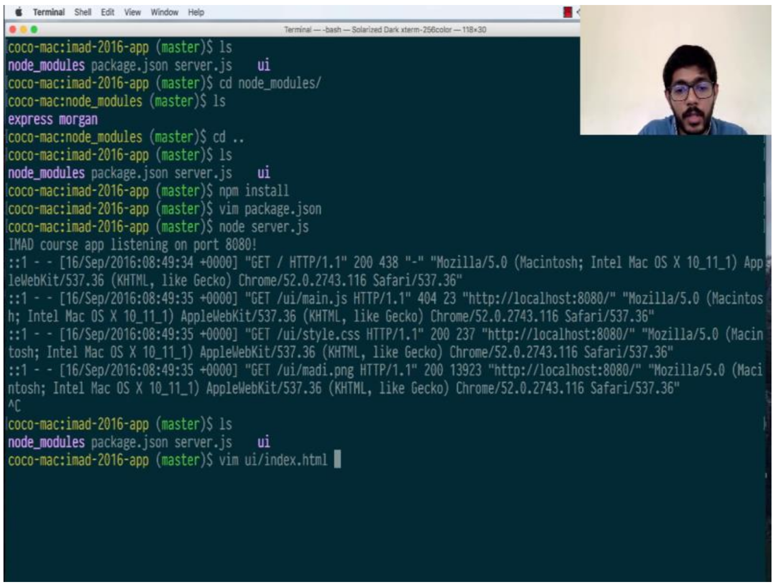
Let us head to the browser. Let us go to local host 8080.

(Refer Slide Time: 19:29)


Hi! I am Tanmai, and this is my first webapp.

And here you can see that our app is running, and this is the app that we changed.

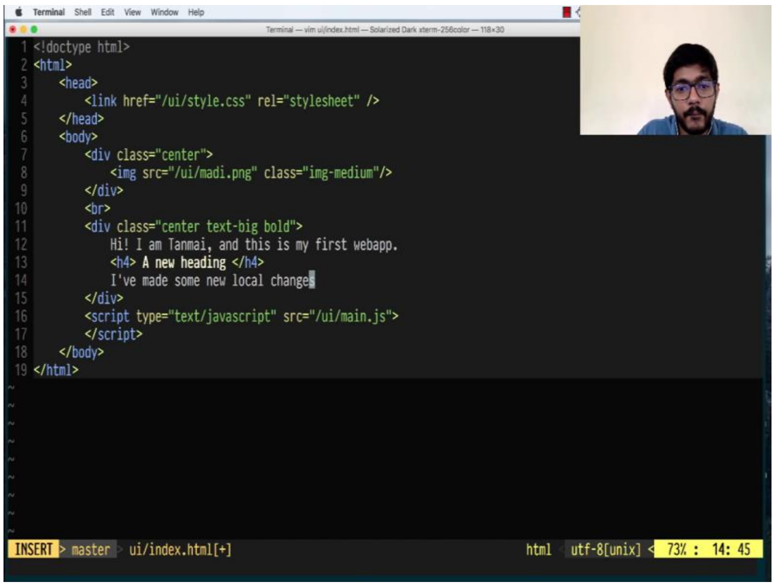
(Refer Slide Time: 19:34)



```
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ cd node_modules/
coco-mac:node_modules (master)$ ls
express morgan
coco-mac:node_modules (master)$ cd ..
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ npm install
coco-mac:imad-2016-app (master)$ vim package.json
coco-mac:imad-2016-app (master)$ node server.js
IMAD course app listening on port 8080!
::1 - - [16/Sep/2016:08:49:34 +0000] "GET / HTTP/1.1" 200 438 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_1) App
leWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
::1 - - [16/Sep/2016:08:49:35 +0000] "GET /ui/main.js HTTP/1.1" 404 23 "http://localhost:8080/" "Mozilla/5.0 (Macintos
h; Intel Mac OS X 10_11_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
::1 - - [16/Sep/2016:08:49:35 +0000] "GET /ui/style.css HTTP/1.1" 200 237 "http://localhost:8080/" "Mozilla/5.0 (Macin
tosh; Intel Mac OS X 10_11_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
::1 - - [16/Sep/2016:08:49:35 +0000] "GET /ui/madi.png HTTP/1.1" 200 13923 "http://localhost:8080/" "Mozilla/5.0 (Maci
ntosh; Intel Mac OS X 10_11_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
^C
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ vim ui/index.html
```

So let us try to make some changes. Let us go to the UI folder index dot HTML.

(Refer Slide Time: 19:39)

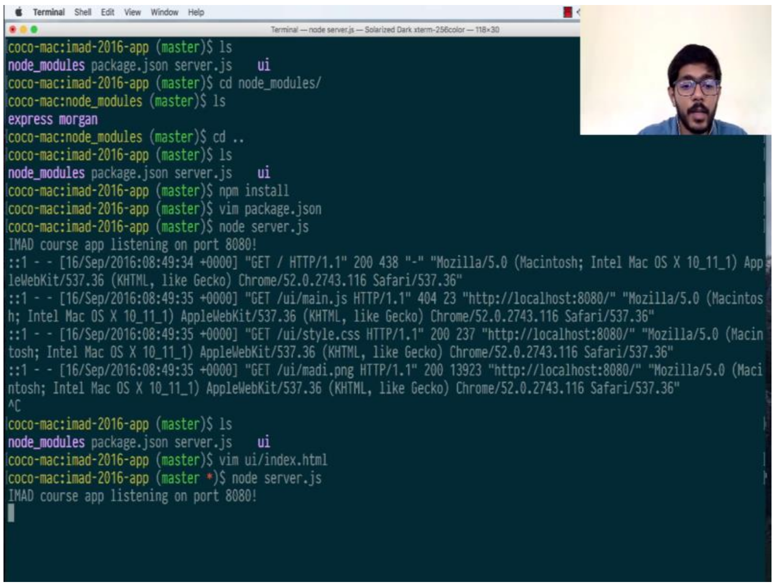


```
<!doctype html>
<html>
    <head>
        <link href="/ui/style.css" rel="stylesheet" />
    </head>
    <body>
        <div class="center">
            <img src="/ui/madi.png" class="img-medium"/>
        </div>
        <br>
        <div class="center text-big bold">
            Hi! I am Tanmai, and this is my first webapp.
            <h4> A new heading </h4>
            I've made some new local changes
        </div>
        <script type="text/javascript" src="/ui/main.js">
        </script>
    </body>
</html>
```

```
INSERT > master  ui/index.html[+]                    html  utf-8[unix]  73% :  14: 45
```

And let us say right, so I saved that.

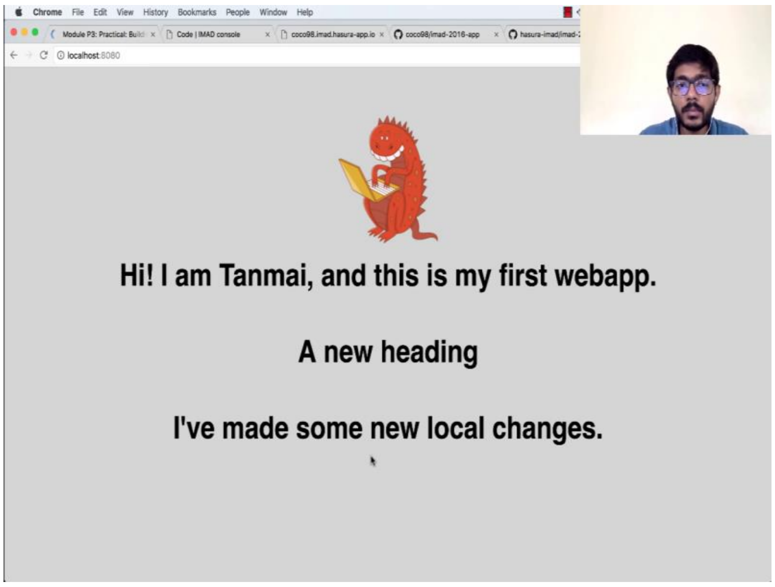
(Refer Slide Time: 20:00)



```
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ cd node_modules/
coco-mac:node_modules (master)$ ls
express morgan
coco-mac:node_modules (master)$ cd ..
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ npm install
coco-mac:imad-2016-app (master)$ vim package.json
coco-mac:imad-2016-app (master)$ node server.js
IMAD course app listening on port 8080!
::1 - - [16/Sep/2016:08:49:34 +0000] "GET / HTTP/1.1" 200 438 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_1) App
leWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
::1 - - [16/Sep/2016:08:49:35 +0000] "GET /ui/main.js HTTP/1.1" 404 23 "http://localhost:8080/" "Mozilla/5.0 (Macintos
h; Intel Mac OS X 10_11_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
::1 - - [16/Sep/2016:08:49:35 +0000] "GET /ui/style.css HTTP/1.1" 200 237 "http://localhost:8080/" "Mozilla/5.0 (Macin
tosh; Intel Mac OS X 10_11_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
::1 - - [16/Sep/2016:08:49:35 +0000] "GET /ui/madi.png HTTP/1.1" 200 13923 "http://localhost:8080/" "Mozilla/5.0 (Maci
ntosh; Intel Mac OS X 10_11_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
^C
coco-mac:imad-2016-app (master)$ ls
node_modules package.json server.js    ui
coco-mac:imad-2016-app (master)$ vim ui/index.html
coco-mac:imad-2016-app (master *)$ node server.js
IMAD course app listening on port 8080!
```

Once again I use the editor vim you guys can use the editor sublime or atom if it is more comfortable to you. I would highly recommend both sublime and atom these are fairly friendly text editors. You can just double click on the file and edit it and save it and everything will be the exact same. There is no need to be worried or confused about getting vim to work on your machines. Now, once you have the setup done, you can just run node server dot js right.

(Refer Slide Time: 20:29)



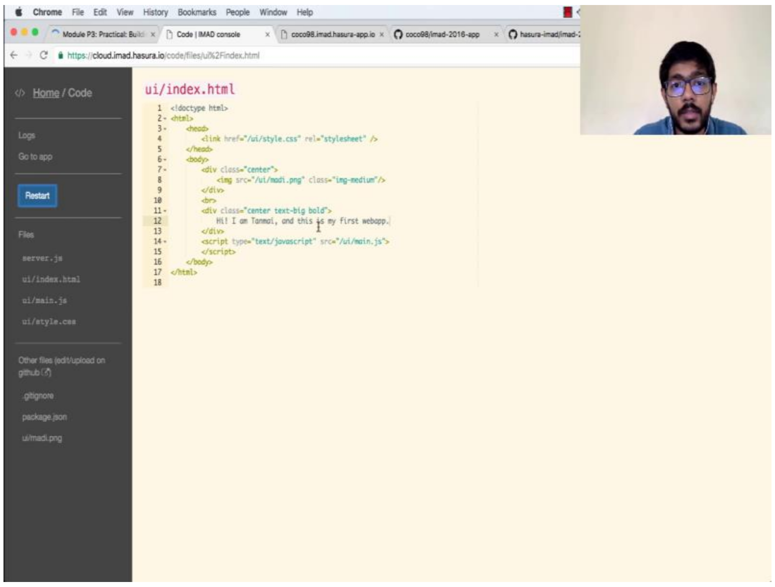
Hi! I am Tanmai, and this is my first webapp.

A new heading

I've made some new local changes.

We have restarted our server and you can see there some new changes have come right.

(Refer Slide Time: 20:43)

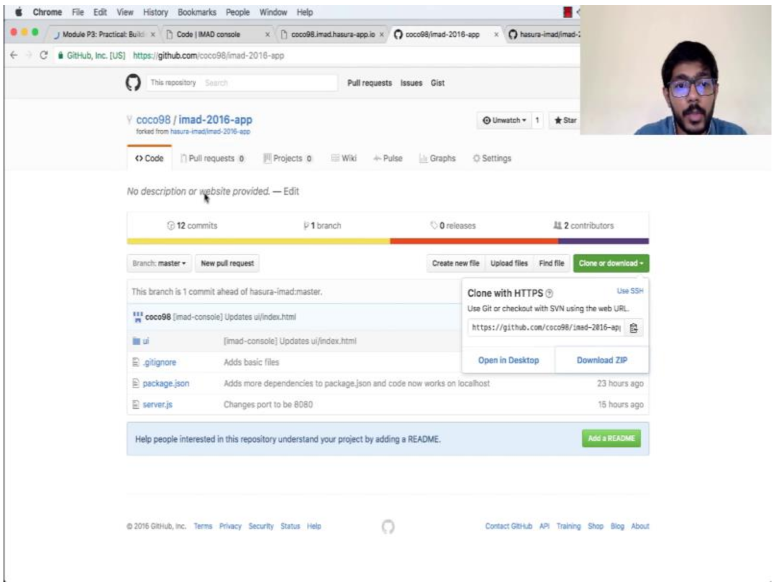


ui/index.html

```
<!doctype html>
<html>
    <head>
        <link href="/ui/style.css" rel="stylesheet" />
    </head>
    <body>
        <div class="center">
            <img src="/ui/madi.png" class="img-medium"/>
        </div>
        <br>
        <div class="center text-big bold">
            Hi! I am Tanmai, and this is my first webapp.
        </div>
        <script type="text/javascript" src="/ui/main.js">
        </script>
    </body>
</html>
```

Let us save these changes right so that, just like how when we edited code on our console; right.

(Refer Slide Time: 20:46)



coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

No description or website provided. — Edit

12 commits | 1 branch | 0 releases | 2 contributors

This branch is 1 commit ahead of hasura-imad:master.

coco98 [imad-console] Updates ui/index.html

ui	[imad-console] Updates ui/index.html	
.gitignore	Adds basic files	
package.json	Adds more dependencies to package.json and code now works on localhost	23 hours ago
server.js	Changes port to be 8080	15 hours ago

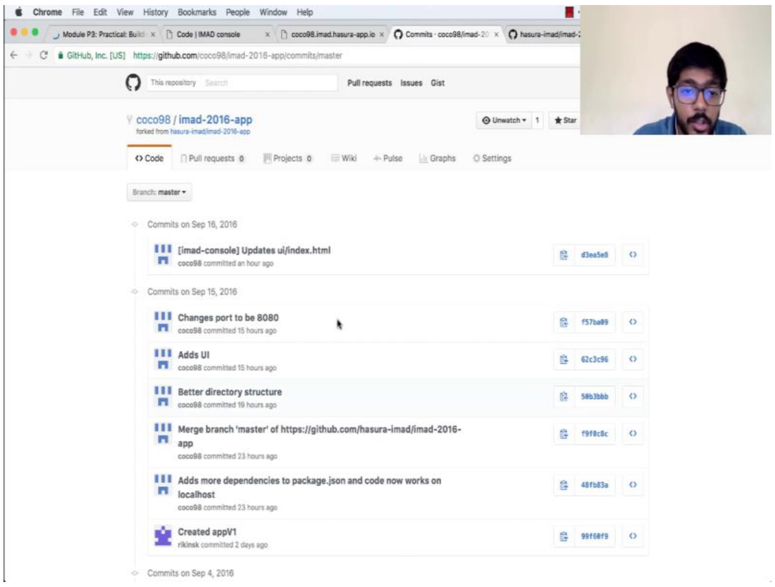
Clone with HTTPS
Use Git or checkout with SVN using the web URL.
https://github.com/coco98/imad-2016-ap

Help people interested in this repository understand your project by adding a README.

Chrome File Edit View History Bookmarks People Window Help
https://github.com/coco98/imad-2016-app
Pull requests Issues Gist
Unwatch 1 Star
Code Pull requests 0 Projects 0 Wiki Pulse Graphs Settings
Branch: master
New pull request
Create new file Upload files Find file Clone or download
Use SSH
Open in Desktop Download ZIP
Add a README
© 2016 GitHub, Inc. Terms Privacy Security Status Help
Contact GitHub API Training Shop Blog About

And it got saved to github which we saw.

(Refer Slide Time: 20:50)



coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

Commits on Sep 16, 2016

[imad-console] Updates ui/index.html coco98 committed an hour ago	d3ea5e5

Commits on Sep 15, 2016

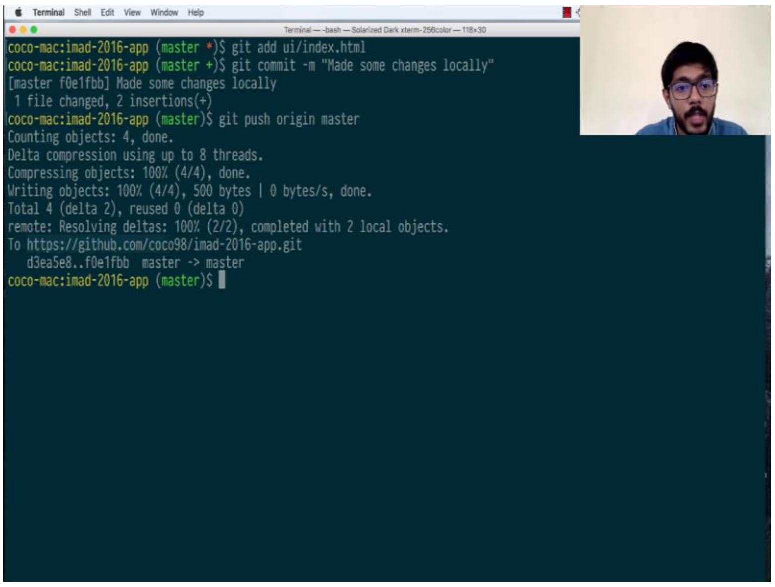
Changes port to be 8080 coco98 committed 15 hours ago	f57ba09
Adds UI coco98 committed 15 hours ago	62c3c96
Better directory structure coco98 committed 19 hours ago	58b3bbb
Merge branch 'master' of https://github.com/hasura-imad/imad-2016-app coco98 committed 23 hours ago	f9f0c8c
Adds more dependencies to package.json and code now works on localhost coco98 committed 23 hours ago	48fb83a
Created appV1 rikinsk committed 2 days ago	99f60f9

Commits on Sep 4, 2016

Chrome File Edit View History Bookmarks People Window Help
https://github.com/coco98/imad-2016-app/commits/master
Pull requests Issues Gist
Unwatch 1 Star
Code Pull requests 0 Projects 0 Wiki Pulse Graphs Settings
Branch: master

In which we saw in the comment history, we also want to save our code.

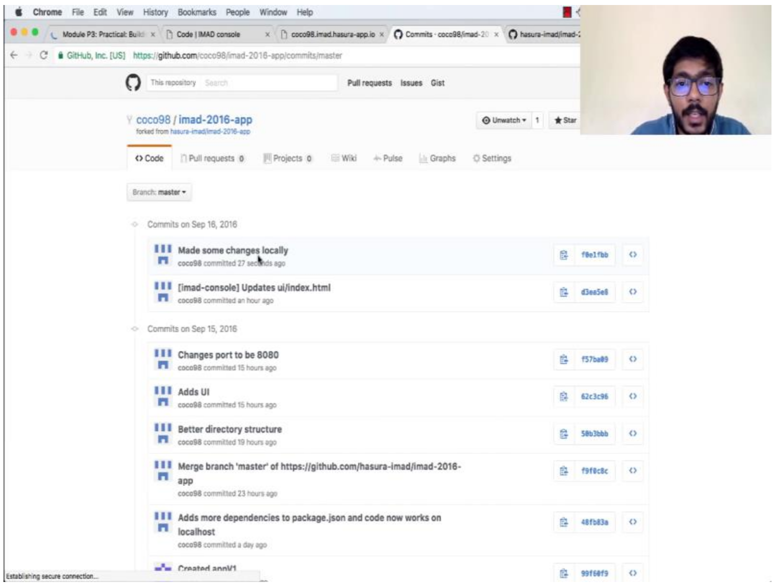
(Refer Slide Time: 20:52)


Terminal Shell Edit View Window Help
Terminal — -bash — Solarized Dark xterm-256color — 118×30

```
coco-mac:imad-2016-app (master *)$ git add ui/index.html
coco-mac:imad-2016-app (master +)$ git commit -m "Made some changes locally"
[master f0e1fbb] Made some changes locally
 1 file changed, 2 insertions(+)
coco-mac:imad-2016-app (master)$ git push origin master
Counting objects: 4, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 500 bytes | 0 bytes/s, done.
Total 4 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/coco98/imad-2016-app.git
   d3ea5e8..f0e1fbb  master -> master
coco-mac:imad-2016-app (master)$ 
```

Let us do git add ui slash index dot html, because we have made changes to that and we want to add it to our next commit. Similarly, I can add more files that have been changed. Once I am done adding the files that are changed, I say commit and I say minus m to send a commit message, which is made some changes locally and once I make these changes I will push it back to the central depository which is on github dot com. So, I say git push origin master and it is saying that it is done to https github dot com slash coco 98 imad 2016 app dot git right it is saying that it is pushed.

(Refer Slide Time: 21:13)


Chrome File Edit View History Bookmarks People Window Help
Module P3: Practical: Buil
Code | IMAD console
coco98.imad.hasura-app.io
Commits · coco98/imad-20
hasura-imad/imad-
GitHub, Inc. [US] https://github.com/coco98/imad-2016-app/commits/master
This repository Search
Pull requests Issues Gist
Unwatch 1
Star
Code
Pull requests 0
Projects 0
Wiki
Pulse
Graphs
Settings
Branch: master
Establishing secure connection...

coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

Commits on Sep 16, 2016

Made some changes locally
coco98 committed 27 seconds ago — f0e1fbb

[imad-console] Updates ui/index.html
coco98 committed an hour ago — d3ea5e8

Commits on Sep 15, 2016

Changes port to be 8080
coco98 committed 15 hours ago — f57ba09

Adds UI
coco98 committed 15 hours ago — 62c3c96

Better directory structure
coco98 committed 19 hours ago — 50b3bbb

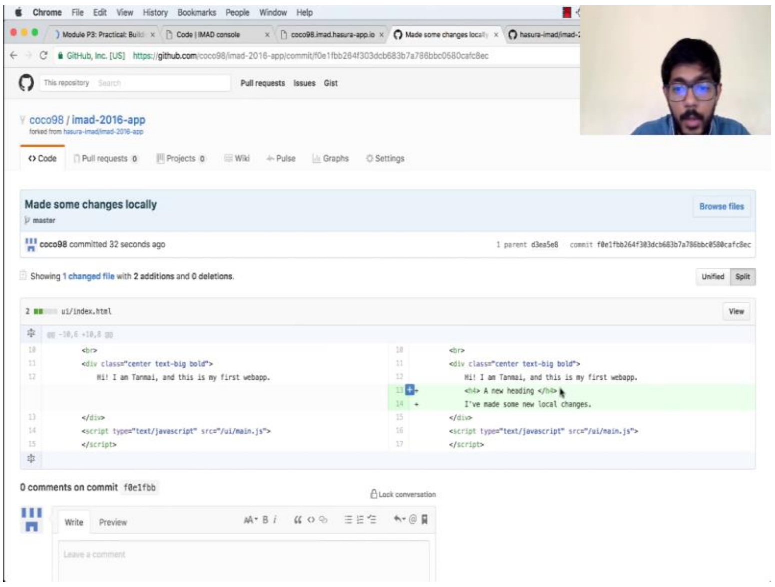
Merge branch 'master' of https://github.com/hasura-imad/imad-2016-app
coco98 committed 23 hours ago — f9f0c8c

Adds more dependencies to package.json and code now works on localhost
coco98 committed a day ago — 48fb83a

Created appV1 — 99f60f9

Let us go back and refresh this page and you can see that the new changes have come up here.

(Refer Slide Time: 21:50)


Chrome File Edit View History Bookmarks People Window Help
https://github.com/coco98/imad-2016-app/commit/f0e1fbb264f303dcb683b7a786bbc0580cafc8ec
Pull requests Issues Gist
coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app
Code Pull requests 0 Projects 0 Wiki Pulse Graphs Settings
Browse files
Unified Split
View
Write Preview
Leave a comment
Lock conversation

Made some changes locally
master

coco98 committed 32 seconds ago   1 parent d3ea5e8   commit f0e1fbb264f303dcb683b7a786bbc0580cafc8ec

Showing 1 changed file with 2 additions and 0 deletions.

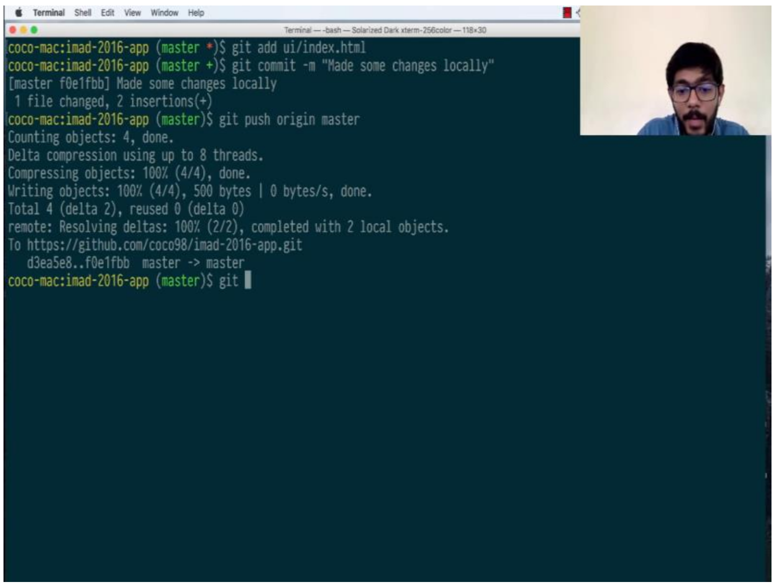
2 ui/index.html

```
@@ -10,6 +10,8 @@
        <br>                                                    <br>
        <div class="center text-big bold">                      <div class="center text-big bold">
            Hi! I am Tanmai, and this is my first webapp.           Hi! I am Tanmai, and this is my first webapp.
                                                              +     <h4> A new heading </h4>
                                                              +     I've made some new local changes.
        </div>                                                  </div>
        <script type="text/javascript" src="/ui/main.js">       <script type="text/javascript" src="/ui/main.js">
        </script>                                               </script>
```

0 comments on commit f0e1fbb

And if you click on this commit again you will be able to visualize what changes you can see here; incidentally github.com powered by git as well.

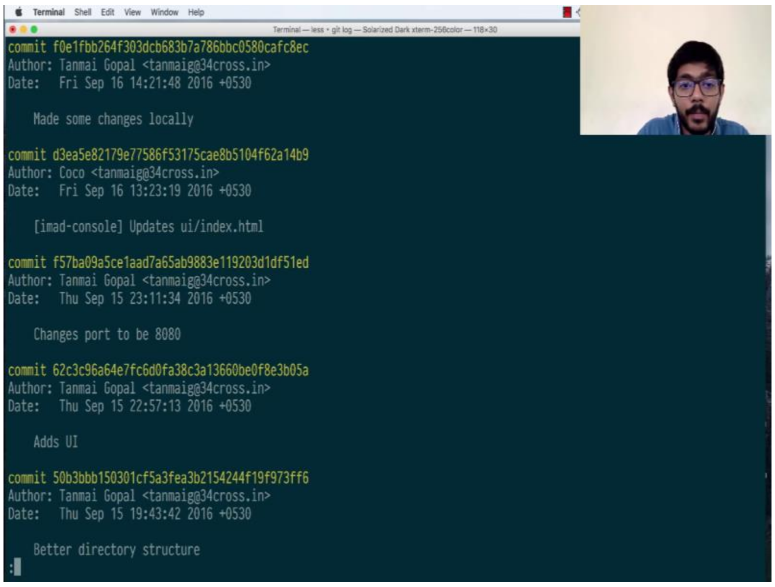
(Refer Slide Time: 21:57)


Terminal Shell Edit View Window Help
Terminal — -bash — Solarized Dark xterm-256color — 118×30

```
coco-mac:imad-2016-app (master *)$ git add ui/index.html
coco-mac:imad-2016-app (master +)$ git commit -m "Made some changes locally"
[master f0e1fbb] Made some changes locally
 1 file changed, 2 insertions(+)
coco-mac:imad-2016-app (master)$ git push origin master
Counting objects: 4, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 500 bytes | 0 bytes/s, done.
Total 4 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/coco98/imad-2016-app.git
   d3ea5e8..f0e1fbb  master -> master
coco-mac:imad-2016-app (master)$ git
```

So, you can run the same command here to see what changes have been made by doing a git log.

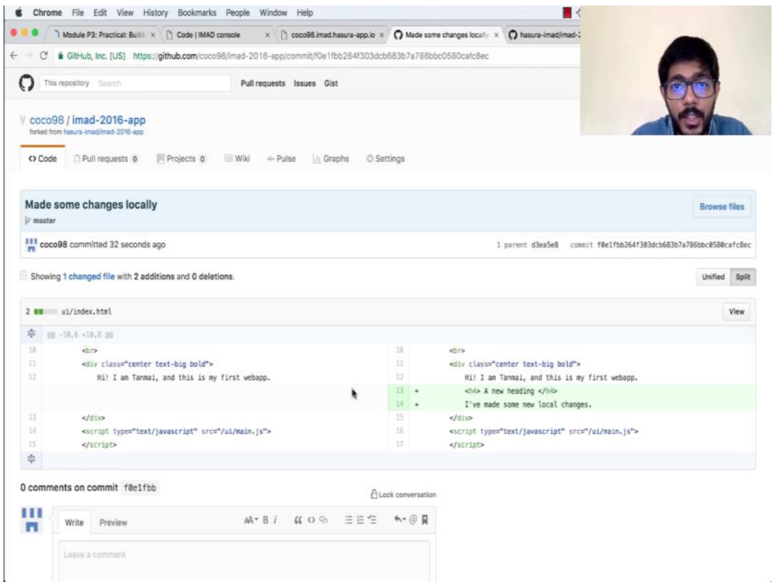
(Refer Slide Time: 22:02)



```
commit f0e1fbb264f303dcb683b7a786bbc0580cafc8ec
Author: Tanmai Gopal <tanmaig@34cross.in>
Date:   Fri Sep 16 14:21:48 2016 +0530

    Made some changes locally

commit d3ea5e82179e77586f53175cae8b5104f62a14b9
Author: Coco <tanmaig@34cross.in>
Date:   Fri Sep 16 13:23:19 2016 +0530

    [imad-console] Updates ui/index.html

commit f57ba09a5ce1aad7a65ab9883e119203d1df51ed
Author: Tanmai Gopal <tanmaig@34cross.in>
Date:   Thu Sep 15 23:11:34 2016 +0530

    Changes port to be 8080

commit 62c3c96a64e7fc6d0fa38c3a13660be0f8e3b05a
Author: Tanmai Gopal <tanmaig@34cross.in>
Date:   Thu Sep 15 22:57:13 2016 +0530

    Adds UI

commit 50b3bbb150301cf5a3fea3b2154244f19f973ff6
Author: Tanmai Gopal <tanmaig@34cross.in>
Date:   Thu Sep 15 19:43:42 2016 +0530

    Better directory structure
:
```

And you can see the commit history although not is nicely in visually in a manner that is as usually appealing as it is on github.com. You can see that I have made some changes locally, and before that the imad console on our behalf updated the ui index.

(Refer Slide Time: 22:18)



coco98 / imad-2016-app
forked from hasura-imad/imad-2016-app

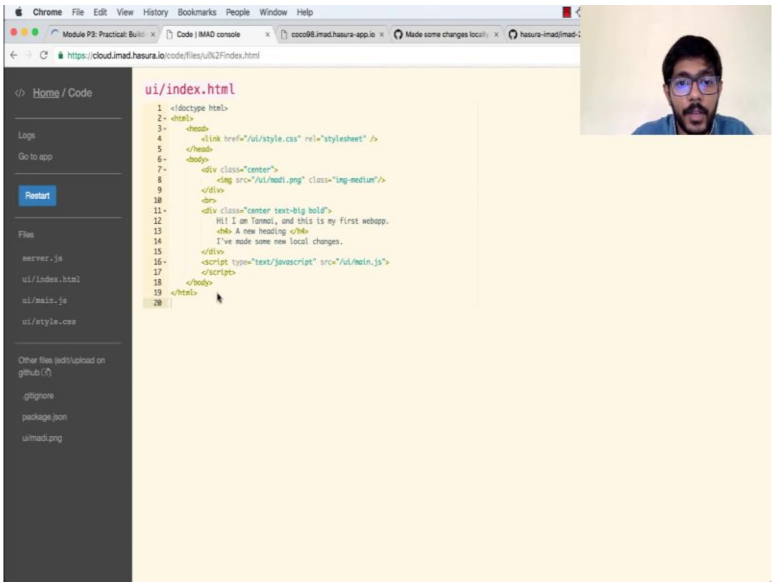
Made some changes locally
master
coco98 committed 32 seconds ago   1 parent d3ea5e8   commit f0e1fbb264f303dcb683b7a786bbc0580cafc8ec

Showing 1 changed file with 2 additions and 0 deletions.

```
2 ui/index.html
@@ -10,6 +10,8 @@
          <br>                                                      <br>
          <div class="center text-big bold">                        <div class="center text-big bold">
             Hi! I am Tanmai, and this is my first webapp.             Hi! I am Tanmai, and this is my first webapp.
                                                                  +    <h4> A new heading </h4>
                                                                  +    I've made some new local changes.
          </div>                                                    </div>
          <script type="text/javascript" src="/ui/main.js">         <script type="text/javascript" src="/ui/main.js">
          </script>                                                 </script>
```

0 comments on commit f0e1fbb

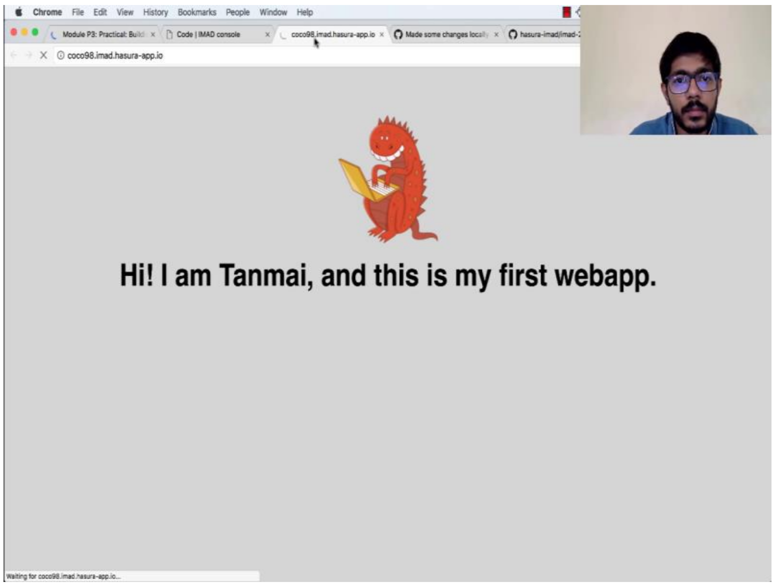
(Refer Slide Time: 22:20)


ui/index.html

```html
<!doctype html>
<html>
    <head>
        <link href="/ui/style.css" rel="stylesheet" />
    </head>
    <body>
        <div class="center">
            <img src="/ui/madi.png" class="img-medium"/>
        </div>
        <br>
        <div class="center text-big bold">
            Hi! I am Tanmai, and this is my first webapp.
            <h4> A new heading </h4>
            I've made some new local changes.
        </div>
        <script type="text/javascript" src="/ui/main.js">
        </script>
    </body>
</html>
```

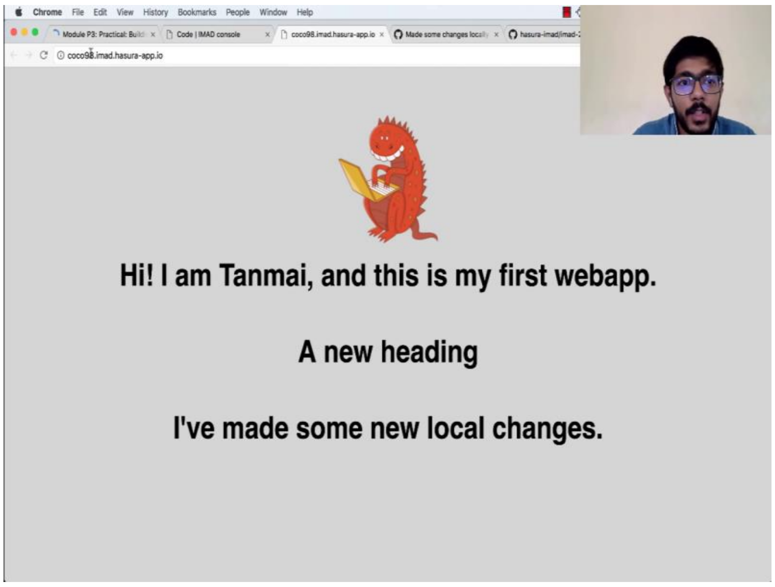
Once the changes are here in github, we can even go back to our console and we can refresh this page, and you will have to wait a while it fetches the files. And you can see that this the console is also refreshed itself, because it is also fetching your files from your github account your github account is the essential place where all changes have been kept in this example. And once this is done, I can click on restart again.

(Refer Slide Time: 22:46)


Hi! I am Tanmai, and this is my first webapp.

And if I restart my server, I should be able to see my changes on the server as while. Let us give it a while.

(Refer Slide Time: 23:04)

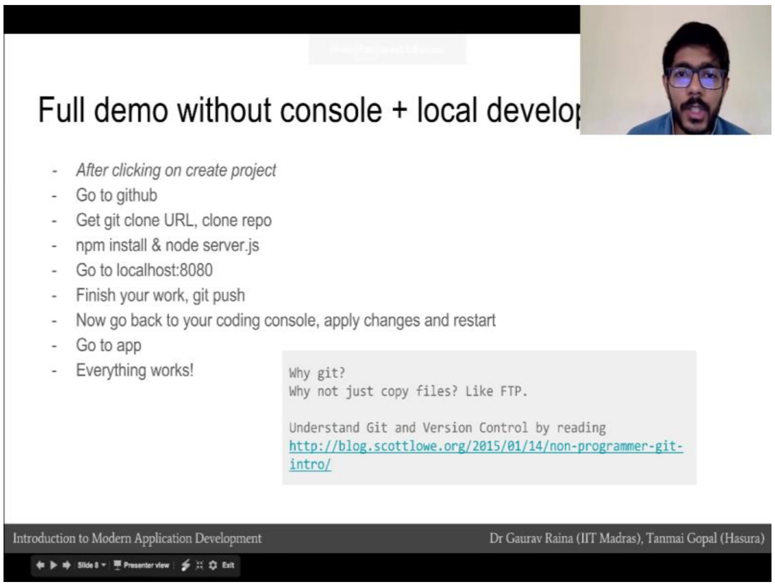

Hi! I am Tanmai, and this is my first webapp.

A new heading

I've made some new local changes.

And you can see that the new changes have come up here right. This is exactly the same what I had on a local host, and now I have it public on the web, so that anybody else can also view these changes.

(Refer Slide Time: 23:15)


Full demo without console + local develop

- *After clicking on create project*
- Go to github
- Get git clone URL, clone repo
- npm install & node server.js
- Go to localhost:8080
- Finish your work, git push
- Now go back to your coding console, apply changes and restart
- Go to app
- Everything works!

```
Why git?
Why not just copy files? Like FTP.

Understand Git and Version Control by reading
http://blog.scottlowe.org/2015/01/14/non-programmer-git-intro/
```

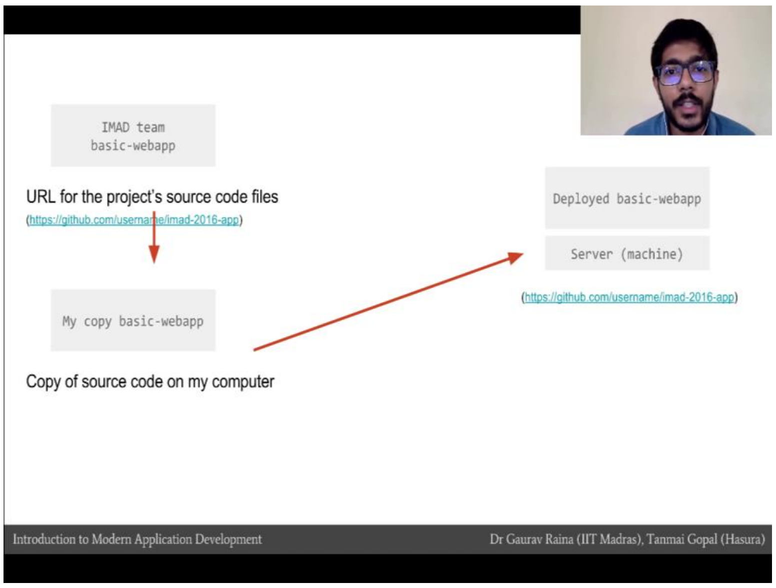
Introduction to Modern Application Development Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

So, why did we use git, why are we using github.com, why are we just copying files, why did we just not copy a zip file and use that to create a copy of the webapp that we wanted to use? Why did we just not copy a zip file and then save it on our local machines and then once you made the changes, create a new zip file and then upload that zip file somewhere, we can we can collaborate on for example, on Google drive or even on drop box.

The reason why we using git is because of all those little features that I showed you, I showed you a commit history I showed you what changes have been made in every single commit and git has the power to let us role back to a particular commit. By if we make any changes or we make any changes that we want to discard, or we want to pull changes from an earlier commit, all of these things are possible with git. And doing this is called versioning your code and git is an example of a vcs one of the most popular vcs a version control system is out that today.

Just an interesting fact git was also invented by the inventor of Linux Linus Tarwals and he invented both Linux and git. And he jokes that his named both the pieces of software after himself. You can read more about git and version control in general by heading to this link that I have mentioned on the slides. And you should read up and understand because git is a powerful piece of software that is very, very useful in all kinds of software engineering.

(Refer Slide Time: 24:41)


IMAD team
basic-webapp
URL for the project's source code files
(https://github.com/username/imad-2016-app)
My copy basic-webapp
Copy of source code on my computer
Deployed basic-webapp
Server (machine)
(https://github.com/username/imad-2016-app)
Introduction to Modern Application Development
Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

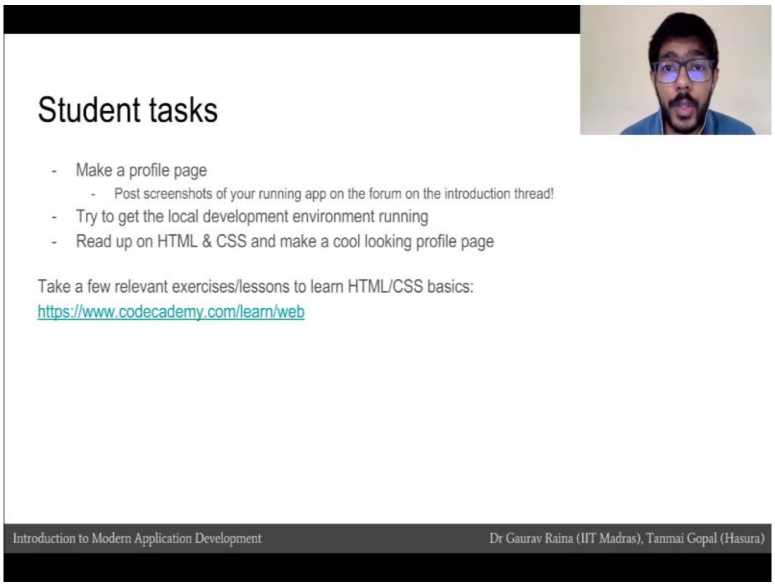
Let us just quickly recap because we covered a lot of complicated concepts. We have a basic webapp with some source code files that has been created by the IMAD team. These were copied when you clicked on the create project button these were copied and a new github project was created automatically for you which was accessible on github dot com slash username slash imad 2016 app. We copied those files on to our local machine by using the command git clone, and we got them on our local machine.

And then after that we pushed it back to our github project; and after pushing it back, when we deployed it those changes were deployed on to our server. And effectively what we were able to do was deploy source code from our computer to our server via github. This might not be how you typically deploy a web server in a more typical setup, you might actually end up copying files, because you have your own server machine you might actually copy files or copy your source code files from your local machine to your server. This is not a recommended practise.

It is always best to use git to or any other version control systems or version your code and then only deploy versions of your code on the server, so that no random files are or mistakes are copied to your server. Again it is also important to understand that everything all the components that we used are fairly optional.

For example, we used github dot com for this particular, we are using github dot com for this course, but you might as well be using bitbucket dot org or gitlab dot com which is all alternative pieces of software that do the same thing. The same thing here being hosting a git project or on the web right; git is a separate piece of software, it is not related to github, it is a separate piece of software that runs on machines. So, it is running on the github server, it is running on our computer, it is running on the IMAD server.

(Refer Slide Time: 26:32)



Student tasks

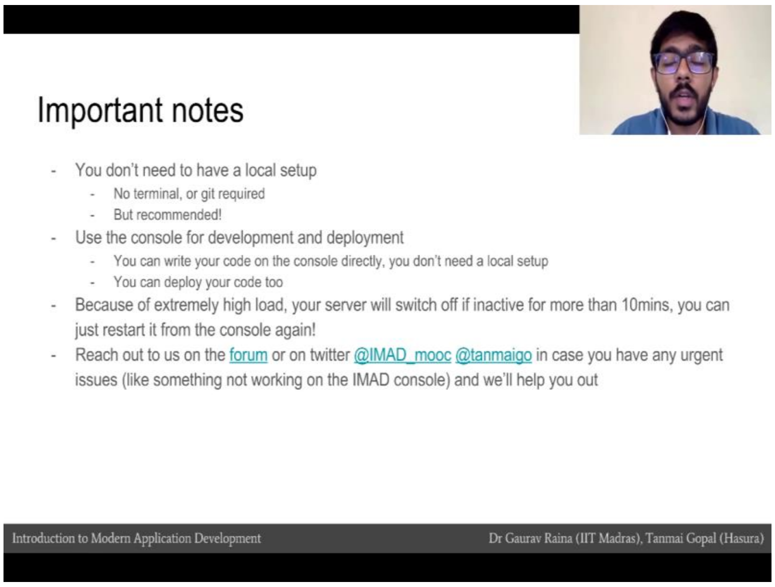
- Make a profile page
    - Post screenshots of your running app on the forum on the introduction thread!
- Try to get the local development environment running
- Read up on HTML & CSS and make a cool looking profile page

Take a few relevant exercises/lessons to learn HTML/CSS basics:
https://www.codecademy.com/learn/web

Introduction to Modern Application Development   Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

What you guys should do next is go on to the console, create your project, and start trying to write some code modify bits in pieces of code keep experimenting and read up a little bit about HTML and CSS basics to try to modify it. And try to make a profile page, you guys can make some profile make a profile page that describes yourself and what you do and post the screenshots of that on the forum.

(Refer Slide Time: 26:55)



Important notes

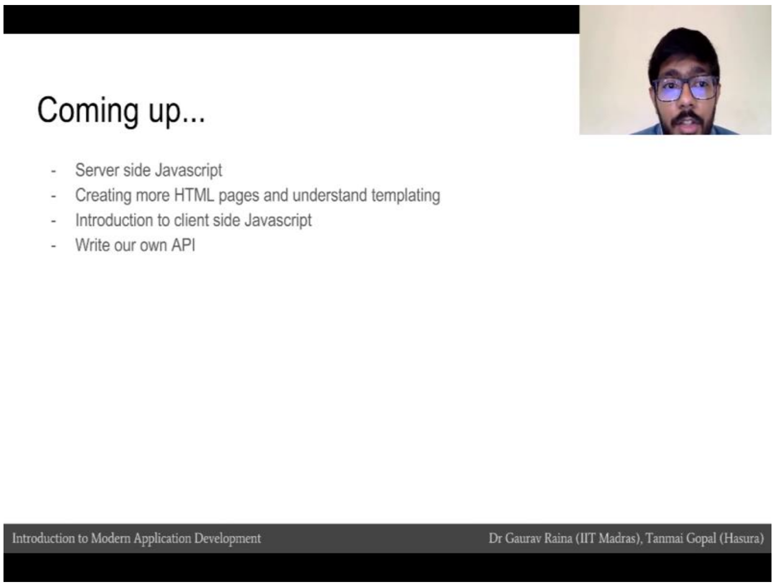
- You don't need to have a local setup
  - No terminal, or git required
  - But recommended!
- Use the console for development and deployment
  - You can write your code on the console directly, you don't need a local setup
  - You can deploy your code too
- Because of extremely high load, your server will switch off if inactive for more than 10mins, you can just restart it from the console again!
- Reach out to us on the forum or on twitter @IMAD_mooc @tanmaigo in case you have any urgent issues (like something not working on the IMAD console) and we'll help you out

Introduction to Modern Application Development Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

You do not need to have the local setup portion to do the basic app development and deployment that is required that is strictly optional. However, it is recommended because if you are going to go on to do slightly more advanced things, and become a developer you will need to have a local setup ready. You can use the IMAD console for development and deployment you can write your code directly on IMAD.

And you do not have to be worried about your changes being lost because all your changes as soon as you click on co commit are saved to github, because of the high load that we will experienced during this course your server might switch off due to inactivity if it is more if it is inactive for more than 10 minutes, you can always restart it just by going back to your console and clicking on restart. In case you have any issues with the forum, in case you have any issues with the console, please feel free to reach out to us directly on the forum or on a twitter handles. And we will make sure that we fix it for you.

(Refer Slide Time: 27:49)



Coming up...

- Server side Javascript
- Creating more HTML pages and understand templating
- Introduction to client side Javascript
- Write our own API

Introduction to Modern Application Development   Dr Gaurav Raina (IIT Madras), Tanmai Gopal (Hasura)

In the next sessions, we will be looking at server side javascript, and understanding how HTML works, we will also be looking at client side javascript. So, although we have not introduced what the source code files are, and how they are working, we will be getting into that from the very next lecture.