Welcome to WDD - 330!

* What you need to know for week 1
* Office Hours
* Expectations
* Teams
* Announcements
* How is grading done

The rest of this email is a lot of bullet points and pretty cut and dried. Let me start with something a little warmer. This is a fun class, or at least it can be. You get to explore JavaScript which is the *lingua franca* of the computing world. Have some fun with it. It is a very powerful language. Java and C# may come and go. Python will rise and fall but JavaScript will be here for a long time. It is the cockroach of programming languages, ugly at times but very resilient. There are some very jazzy things (dare I say groovy, without showing my age?) to be done with JS. Explore it, don’t do the assignment and stop. Press on and try something new. You are at BYU-Idaho not to get a grade but to get an education. Your grades might get you a job but your ability to explore and experiment will get you your promotions.

Let’s get the week started and we will get into the weeds soon enough.

# Week 1

* STOP and respond to this request – When you can meet with your team? [Use this link](https://docs.google.com/spreadsheets/d/1g-o0Ldapnu5fiMSgEXTG_3MopmDGLyc0VddFuGSexYI/edit?usp=sharing) to find a spreadsheet with the days of the week and the times of the day you can meet. Add your name to the row that fits your availability. If you see that your row is not getting more that 4 people move to another time. If a team has more than 7 or 8 then you can, or I will, split the team into two smaller groups. I have two sections of this class which means we have the ability to accommodate your schedule a little more, but you will not be able to use the team discussions in Canvas. Not a showstopper, just be aware.
* Make sure you do your weekly reading assignment. The first few weeks will probably be review for most of you. Create the examples on your own that you find in the reading material. Expand on them. Have some fun with them.
* Your submissions for your weekly assignments will frequently include a web exercise. That means you must have a way of allowing me to test your code. I will not download your code. **Your need a web site of some kind.** See item below for a great suggestion.
* Get your repository set up. I would really recommend you create a GitHub **pages** repository. That gives you a ‘web site’ as well as your code repository. It is as simple as creating a repo named so: **yourUsername.github.io** (for instance this is mine <gtjames.github.io> go ahead click on it. See what it does). An even better solution is to have your own web site, this is your field after all.
* Week 1 is not terribly challenging.
  + Bookmark the websites mentioned in the reading
  + Create a website for your assignments
  + Set up the repo you will be using (github.io is a great suggestion)
  + Practice the simple examples in the reading
  + Read ahead to weeks 2 and 3 to see what comes next

# Office Hours

I will have offices hours on Tuesday and Thursday evening for the first few weeks. After that you will be working more in your teams and I will make it a point to get by frequently to say hi and answer questions.

For the next **few** weeks, you can find me at:

<https://byui.zoom.us/my/bpJames>

Tuesday 7:30 PM **Mountain**

Thursday 7:30 PM **Mountain**

I will start the meetings THIS week. See you Tuesday or Thursday. I will post the recordings if you miss the live session.

You can always contact me through Canvas messages or [jamesga@byui.edu](mailto:jamesga@byui.edu)

# Teams

You will find your own teams with mutually agreed upon meeting times. Again, [here is the link](https://docs.google.com/spreadsheets/d/1g-o0Ldapnu5fiMSgEXTG_3MopmDGLyc0VddFuGSexYI/edit?usp=sharing) to coordinate your day and times. You can see that some days there is room for multiple evening teams. Find a slot that works and add your name. I will have to do some adjusting at the end of the week. But this gets us started

Table

Description automatically generated with medium confidence

# Expectations for you and from me

This course is quite different from many other courses you have taken.

* I will not change the due dates. Do not procrastinate.
* I will provide some examples and discussion on most of the topics.
* I have 100s of examples from previous students to share.
* I will join your meetings about every 2-3 weeks to say hi and answer questions
* If you can’t figure something out in your team discussion. Let me help. I will be happy to set up a Zoom meeting to look at your code; after you work on it, research it and check with your team members.
* Most of the course experience (**by design**) is for you to use the resources available to you through online means and in your teams.
* Be prepared to show initiative. Be motivated. Do something EVERY week.
* Stay current with the weekly modules and you will be OK.
* You will be in teams.
* You will have a team assignment most weeks. Do not miss a team meeting.
* If you are engaged with your team, you will solve most of your roadblocks there.

# Communication – Weekly Announcements

I will communicate with you weekly. Just like this. I will post an Announcement and frequently hide a question in the body of the message to see if you are reading. Please email me (jamesga@byui.edu) if you have questions. If we need to have a ‘face-to-face’ conversation to look at some code give me a few days warning and several times you are available so we can coordinate schedules AND explain your difficulty, what you have already done to resolve it.

# Grading

30% comes from your Readings-Prepare/Ponder (9 readings worth 10 points each) and Team Activities (10 activities worth 20 points each)

70% comes from Weekly Reports (12 X 3 points each), Challenge Proposal (1 point), Final Portfolio Submission (100 points)

Which means the Final project submission is almost **40%** of your grade!

Something is due EVERY week. But the biggest assignment is not due until the end of the course. Be thinking about it starting early on. Procrastination is a killer.

Due dates will not be altered. Make sure you submit your work on time.

Bro. James

P.S.

For you Type A folks, here is a heads up on the coming weeks

Week 2 you MUST have a team selected. You will start meeting in your teams. You will share your meeting schedule with me so I can drop by now and then.

Weeks 3 – 10 read the assigned material, meet with your team, do the assigned team project, do the weekly reporting assignments and submissions.

Weeks 11 – 14 build an application that sums up what you have learned. Here is the rubric I use for that final submission.

**Meaningful usage of the following topics**

30% Third party APIs

More than a wimpy Chuck Norris API is not going to cut it (<https://api.chucknorris.io/jokes/random>). There are MANY great APIs for weather, travel, sports, NASA, movies, music, Anime, search, GPS, maps. Check out <https://rapidapi.com/marketplace>. I expect an API should return a good rich JSON result set. Something with an array you can iterate through. Weather for the next 8 days, 10 matching movies, song list for an artist, books from an author, pictures from a Mars rover, sports scores. BTW, if you do this you will also get the points for JSON and it is impossible to do a good job here without some serious JavaScript. APIs+JavaScript+JSON and you already have 80% of the points!!

30% JavaScript

I expect some honest logic going on here. Validating the screen data, looping through an array of JSON data to display to the screen, creating and using events, changing element styles with JS, changing element classes to use different CSS rules.

25% CSS

Use Transforms to round the edges of your input fields, add shadows. Use Transitions to enlarge an input field on focus, and shrink it on blur, wiggle an input field when bad data is entered. Add borders. Don’t go overboard. CSS should subtly add style to a page, not whack you over the head.

20% JSON

JSON is the *lingua franca* of our time. Don’t use it at your peril.

15% Events

Use events to enhance the user experience; increase the size of the input field on focus or add a shadow. React to a button click. Initialized the page with data once the onload event triggers.

If you have CSS rules for pseudo-state-selectors (hover, focus, checked, after, before) I’ll recognize these as events as well.

10% LocalStorage

LS is a nice to have. By design it is local to current device. It is not available as you move around from phone, to tablet, to laptop. But it is nice to know how to implement. It can add value to an app to remember the user from their last visit. Not a lot of points but fairly easy to add to an app.

Yes, this adds up to more than 100%. You can get 90%+ without using APIs. But you can see how biased this is towards using them. They are extremely useful and are ubiquitous in enterprise development.