Unit 2: Digital Applications - Program of Learning (Web Dev)

* <http://www.bsss.act.edu.au/__data/assets/word_doc/0004/454261/Digital_Technologies_A-T-M-V_20-24.docx>
* All of the below are **suggestions** on a possible sequence to deliver using the Digital Technology Course of the BSSS Technology Framework.
* Depth of the topics covered should be dependant on the skills/background of the teacher and students and are **suggestions** of experiences and skills the students could be exposed to.
* They are not expected to be delivered at significant depth unless the teacher/students are willing to do so.

# Web Design PoL

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Wk** | **Topic** | **Fundamental Concepts** | **Possibilities Classroom Activities / Assessment Items** | **Resources** | **Curriculum Links** |
| 1 | Data Formats and Data Driven Websites | * What is the difference between a static and dynamic website? * Why is data driven so important? * What kind of data is useful? * Do we always need an RDBMS? * What are JSON, BLOB and/or CSV data storage objects, how do we build them, and why are they useful? * What is Local Storage? * What are the issues with the management of data and data storage? | * Review data driven sites (social media pages, anything modern really) and contrast them against static sites with collections of pages. * Discuss the significance of data, storing it, and the value of different formats. * Contrast object storage methods against relational storage methods.   **Possible Assessment Task**   * Students construct a simple data-driven website using a JSON, BLOB, CSV, or similar simple data storage object so that the website can import data and then save changes (using either Local Storage or existing external data files) | Local Storage tutorial:  <http://taybenlor.com/2013/08/25/making-a-todolist-app.html>  Using JS to write info to an external file:  <https://www.geeksforgeeks.org/javascript-program-to-write-data-in-a-text-file/> | 2a.  2b.  2c.  2e.  2f.  3b.  3c.  3d.  5a.  5b.  5c.  5d.  6a. |
| 2 |
| 3 |
| 4 |
| 5 | MVC Architecture | * What is the MVC architecture and how does it apply to websites? * How is this different from the previous implementation? * Why is the MVC architecture useful? * How can a design process be applied in the development of an MVC driven application? | * Review the MVC architecture and contrast it against a website structure without a separate controller. * Discuss how the design process can apply to the development of an MVC driven application | MVC Overview:   * <https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm> * <https://www.tutorialsteacher.com/mvc/mvc-architecture> | 1a.  2a.  2b.  2c.  3a.  3c. |
| 6 | Building a Server Part 1 - Python | * What is a server, what do they do and why are they useful? * How can we build a server? * What are the programming principles of python and how do they differ from JavaScript? | * Define a server and discuss its purpose * Overview server construction and how it ties into the MVC architecture * Introduce programming principles in python:   + Data types     - Integer, Character, Float   + Control Structures:     - If Statements     - For Loops     - While Loops   + Data Structures:     - Lists     - Dictionaries   + Functions   + **Brief** overview of Object Oriented programming principles:     - Encapsulation     - Inheritance     - Polymorphism     - Abstraction   **Possible Assessment Task**   * A portfolio of work/reflective journal on the programming principles learnt * A programming exam | Python Tutorials (free):   * In depth python concepts - <https://runestone.academy/runestone/books/published/thinkcspy/index.html> * Python OO - <https://realpython.com/python3-object-oriented-programming/>   Intro Python Tutorials ($30/student/year for all of Grok’s content, free for teachers):  <https://groklearning.com/course/intro-python-1/> and <https://groklearning.com/course/intro-python-2/> | 1b.  2a.  2b.  2c.  2e.  2f.  5a.  5b.  6a. |
| 7 |
| 8 |
| 9 |
| 10 |
| 11 | Building a Server Part 2 - Flask | * What is Flask? * How can we use Flask to implement the MVC architecture for a completed website using an RDBMS or other data storage object? | * Overview MVC architecture in the context of the Flask library as the controller, html as the view and an RDBMS or other data storage object as the model depending on the complexity of the data.   **Possible Assessment Tasks**   * Build a flask server in python and connect it to a view and model. Use a design process to develop the application and justify any design decisions made. * Produce a portfolio of work based on the development of a flask application. | In-depth Flask Tutorial:  <https://blog.miguelgrinberg.com/post/the-flask-mega-tutorial-part-i-hello-world> | Every section of unit content |
| 12 |
| 13 |
| 14 |
| 15 |
| 16 | Catch Up Weeks/ Final Assessments | * Time is left here on the assumption that this is a second semester unit. * If run in Semester 1, more time can be allocated to Python and Flask as needed. | **Possible Assessment Task**   * End of unit Test * Set of Prac. Exercises |  |  |
| 17 |
| 18 |
| 19 |
| 20 |