

Building UIs with Flask

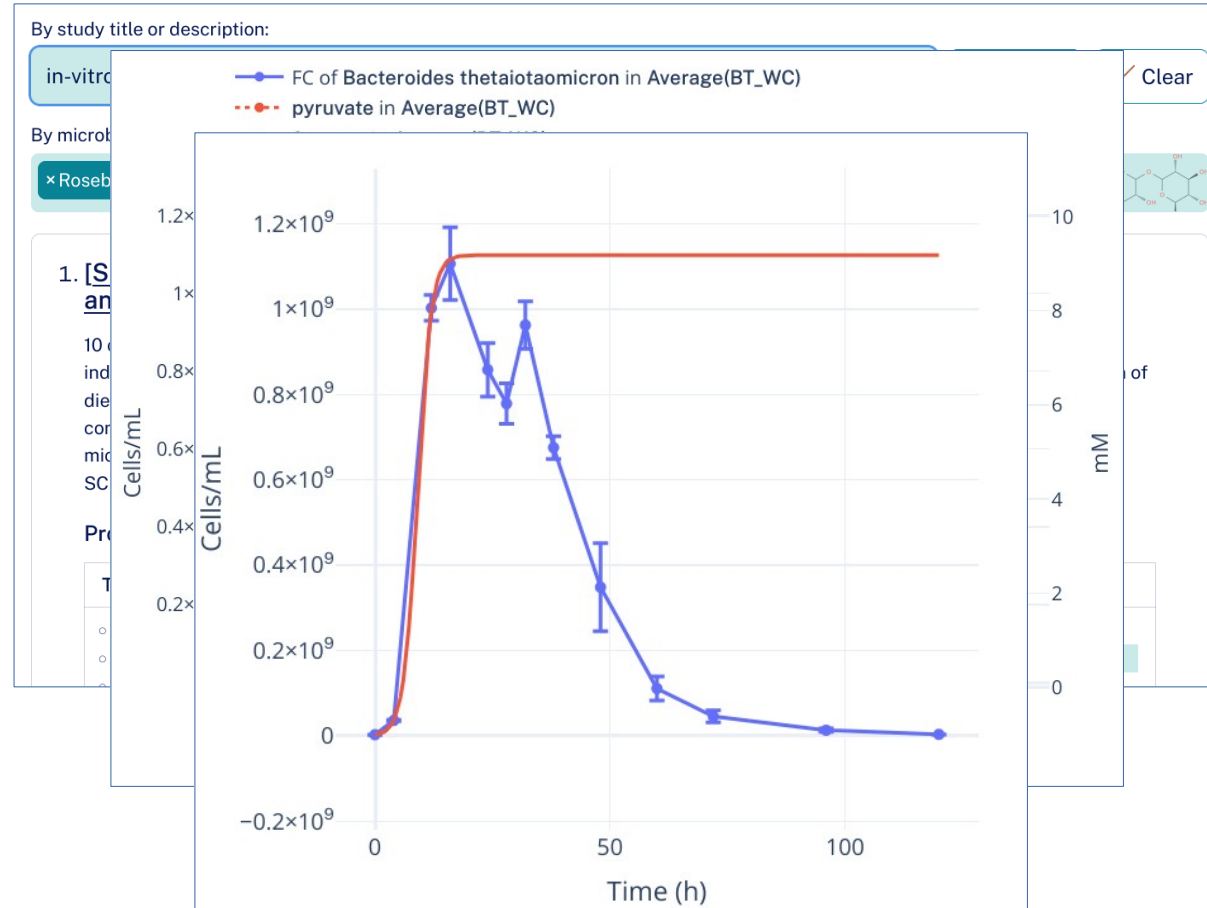


About me

- Andrew Radev
- Used to be a professional software developer
- (... mostly websites in Ruby + Javascript)
- Master of Bionformatics at KU Leuven
- (... a website in Python + Javascript)

μ GrowthDB

- Microbial growth curves
- Searching for studies
- Visualizing data traces
- Modeling
- **Sharing research data**



Getting started

- <https://andrewradev.com/rse-2025>
- Flask skeleton
- Not focusing on the logic
- Important stuff in HTML, CSS, JavaScript
- Practical workflow
- References to documentation

HTML

```
<em>Content</em>
```

```
<strong>Content</strong>
```

```
<a href="https://example.com">Content</a>
```

```
<div class="content-container" id="content1">  
  Content  
</div>
```

```
  
<br />
```

```
  
<br>
```

HTML

- Most important resource: [MDN](#)
- Purpose: describe the structure and meaning of the content, **not** (necessarily) the visuals.
- “Semantic HTML”

Jinja2

The result is: `{{ 1 + 1 }}`.

```
<em>{{ variable }}</em>
<span>{{ collection|length }} items</span>

{% if condition: %}
  <div>Content</div>
{% endif %}

{% for item in collection: %}
  <li>{{ item.property }}</li>
{% endfor %}

{# Comment #}
```

Jinja2

- Main documentation: [RTD](#).
- Built-in filters: [full list](#).
- Mostly like writing python in HTML (though not quite enough like it)

XSS

- “Cross-site scripting”
- Jinja sanitizes everything by default
- Can use the “|safe” filter if we’re sure the input is safe
- Can use the “markupsafe” package directly: [Flask docs](#).

CSS: selectors

```
ul {  
  list-style: lower-roman;  
}
```

```
.entries-list {  
  margin-top: 10px;  
}
```

```
#back-to-top {  
  position: fixed;  
  bottom: 40px;  
  right: 40px;  
}
```

CSS: nesting

```
a.green {  
  color: green;  
  text-decoration: none;  
}  
  
a.green:hover {  
  color: lightgreen;  
}  
  
a.green span {  
  text-decoration: underline;  
}
```

```
a.green {  
  color: green;  
  text-decoration: none;  
  
  &:hover {  
    color: lightgreen;  
  }  
  
  span {  
    text-decoration: underline;  
  }  
}
```

CSS: important concepts

- [Box model](#): padding, border, margin
- [Position](#): relative, absolute, fixed
- [Display](#): inline, block, inline-block, flex
- [Z-index](#): which element is on top
- Full documentation, tutorials etc: [MDN](#).

CSS: flex

- Flex documentation: [MDN](#).
- Tutorial 1: [Flexbox froggy](#).
- Tutorial 2: [Flexbox defence](#).

CSS: colors

- [Tailwind](#): a CSS framework
- Named web colors: [wikipedia](#).

JavaScript

```
if (condition) {  
  console.log("true");  
} else {  
  console.log("false");  
}
```

```
let collection = [1, 2, 3];
```

```
for (let item of collection) {  
  console.log(item);  
}
```

```
let object = { foo: 'bar', bar: 'baz'};  
console.log(object);
```

JavaScript

```
function someName(arg1, arg2) {  
  return arg1 + arg2;  
}
```

```
let someName = function(arg1, arg2) {  
  return arg1 + arg2;  
};
```

```
let someName = (arg1, arg2) => {  
  return arg1 + arg2;  
};
```


JavaScript

```
let someName = (arg1, arg2) => arg1 + arg2;
```

```
let someName = arg1 => arg1 + 10;
```

```
let someName = () => console.log("OK, I guess");
```

JavaScript

- Full reference: [MDN](#).
- Game tutorial 1: [Code Maven](#).
- Game tutorial 2: [Game Maven](#).

More resources

- Already mentioned: MDN, Flask homepage, Jinja RTD
- μ GrowthDB: [source code](#).
- Deployment: [μGrowthDB RTD \(my fork\)](#).
- Systemd: [Arch wiki](#).
- Javascript: [jQuery](#).
- Reach out by searching for “Andrew Radev” on various sites.

Thank you