

Informatics 134

Software User Interfaces
Spring 2023

Mark S. Baldwin

baldwinm@ics.uci.edu

4/19/2023

Agenda

1. Upcoming
2. Callbacks and Events
3. Team Project Outlook and Discussion
4. Team Activity
5. References

Upcoming

Agenda

We are taking attendance now. Teams will be required to report who was in class by the end of the day the class occurred! Use your team Slack channel to let us know who was **missing**. Honor system will be used, but if seats start looking empty, we will take a more rigorous approach.

- Today:
 - Quick lecture (I promise!)
 - Group toolkit discussion
 - T2 Due tonight!!
- Next Week:
 - A1 Due Monday (4/24)

Callbacks and Events

Event Callbacks

User input, events, and action

Differ across programming languages
and toolkits

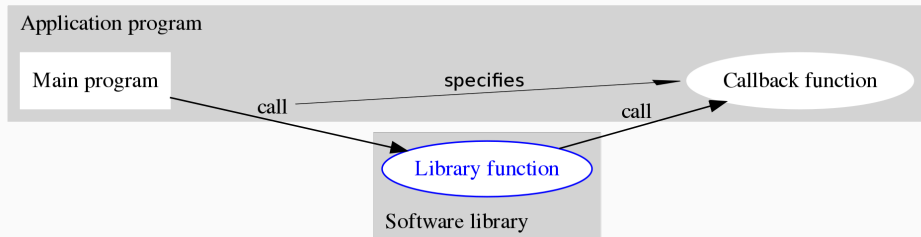
Referred to as: “callbacks”, “event
handlers”, “actions”, and others

User input, events, and action

In the browser...

Input widgets (text, check, button,
heading, div, etc)

Angular's data bindings (e.g., "ng-bind")



[Wikipedia, 2021]

An HTML and Javascript Example

A function is referenced via HTML attribute and *called* when the specified event is performed (a click).

Here the DOM manages the connections for us, but in our Toolkit we are responsible for connecting user input to action.

```
1 function buttonClick(e){  
2     ...do some action  
3 }
```

```
1 <button onclick="buttonClick(this)">...</button>
```

User input, events, and action

In Javascript/Typescript and SVG.js

Callbacks are functions that
we pass as objects

First, we must make the
function anonymous

And because functions are
first-class citizens in
JavaScript...?

```
1  function buttonClick(e){  
2      ...do some action  
3  }  
4  
5  // becomes:  
6  
7  var buttonClick = function(e){  
8      ...do some action  
9  }
```

User input, events, and action

In Javascript/Typescript and SVG.js

We can pass them to other functions as arguments. So...

Callbacks are simply functions that we pass as objects

We can use callbacks to customize the propagation of user input actions

```
1 var buttonClick = function(e){  
2     console.log(e)  
3 }  
4  
5 function MyCallback(action){  
6     action("MyCallback was called")  
7 }  
8  
9 MyCallback(buttonClick)
```

```
1 > "MyCallback was called"
```

User input, events, and action

In Javascript/Typescript and SVG.js

In the example to the right,
what will be the output of a
hover event?

```
1 var w = new MyWidget();  
2 let callback = function(event:any){  
3     console.log("I am being hovered!");  
4 };  
5 w.onHover(callback);  
6
```

```
1 class MyWidget extends Widget{  
2     onHover(callback: { (event?:any): void }):void{  
3         this.attach(callback);  
4     }  
5     ...  
6     hoverState(): void{  
7         this.raise(new EventArgs(this));  
8     }  
9 }
```

User input, events, and action

Why? How do callbacks help us build toolkits?

- Separation of concerns

- Clean up operations

- Pass control to consuming (or calling) code

- Asynchronous operations (promises in JS)

Event Callbacks

We'll cover callbacks and events in more depth next week with the introduction of Individual Assignment 2.

Team Project Outlook and Discussion

Team Project Outlook and Discussion

Goals for the Team Project

- We are focusing on the user interface for this course
- Explore novel or interesting approaches to handling user interaction
- Projects do not need to be end-to-end or fully functional
- If necessary, "Wizard-of-Oz" your final prototype
- Hard code data, 'simulate' envisioned interactions that you can't make function!
- Assessment will emphasize creativity over code and functionality

For Discussion 1

See next page...

Novel Graphical Interface Examples

SpaceTokens

Sliding Widgets 1

Sliding Widgets 2

Contextual Help

Multimodal Tasks

Color Builder

Team Activity

Discussion 1: Requirements

In Class

- As a team, write down the name of the UI toolkit/framework you are using in the spreadsheet
- Write down when it was originally launched
- Describe the type of license it operates under (MIT, BSD, CC)
- Describe who created or backed its creation (Open Source, Facebook, Google)

For Discussion 1

See next page...

Discussion 1: Requirements

Include the data from the spreadsheet along with the following points

- History
 - When was it launched?
 - Why was it launched (*e.g.*, what perceived problem was it created to solve)?
 - What tool(s) are used to develop with it?
 - What OSes does it run on (native)
- Features
 - What programming language(s) does it use?
 - What are its advantages over similar frameworks?
 - What are the trade-offs?
- Sample Code (show, don't implement)
 - A button and corresponding click event.

Discussion 1: Due Date

When should it be due?

References



Wikipedia (2021).

Callback (computer programming).