

Designing Reusable Web Components

Dr. Joonas Lehtinen
Vaadin

@joonaslehtinen

vaadin }>

Agenda

What do we
want to
design?

Q & A



Technology

- HTML5 / Canvas
- Google Web Toolkit
- Vaadin Framework



Designing Web Component
Step-by-step



Goal



Goal


Spreadsheet component XLS upload demo

Upload a XLS file to view it

Choose File no file selected Upload

t2.xls

	1	2	3
41	41.0	1651.0	1.6127688307137333
42	42.0	1764.0	1.6232492903979003
43	43.0	1849.0	1.6334684555795864
44	44.0	1936.0	1.6434526764861872
45	45.0	2025.0	1.6532125137753435
46	46.0	2116.0	1.6627578316815739
47	47.0	2209.0	1.6720978579357173
48	48.0	2304.0	1.6812412373755872
49	49.0	2401.0	1.6901960800285134
50	50.0	2500.0	1.6989700043360185
51	51.0	2601.0	1.7075701760979363
52	52.0	2704.0	1.716003343634799
57	57.0	3249.0	1.7558748556724912
58	58.0	3364.0	1.7634279935629371



<http://jole.virtuallypreinstalled.com/spreadsheet>

```

public class SpreadsheetUploadDemo extends Root implements Receiver
{
    Upload upload = new Upload("Upload a XLS file to view it", this);
    Spreadsheet spreadsheet = new Spreadsheet();

    protected void init(WrappedRequest request) {
        setCaption("Spreadsheet component XLS upload demo");
        VerticalLayout vl = new VerticalLayout();
        vl.setMargin(true);
        vl.setSizeFull();
        vl.addComponent(upload);
        setContent(vl);
        vl.addComponent(spreadsheet);
        vl.setExpandRatio(spreadsheet, 1.0f);
        spreadsheet.setGraphEnabled(true);
    }

    public OutputStream receiveUpload(final String filename, String mimeType) {
        ByteArrayOutputStream baos = new ByteArrayOutputStream() {
            public void close() throws IOException {
                super.close();
                spreadsheet.setCaption(filename);
                spreadsheet.readXLS(new ByteArrayInputStream(this.toByteArray()));
            }
        };
        return baos;
    }
}

```

A large, stylized graphic of the numbers '1' and '2' in a rounded, sans-serif font. The '1' is dark gray and the '2' is blue. The word 'Technology' is written in white across the middle of the '2'.

Technology

HTML



<!doctype html>

Element	Description
<article>	An independent piece of content for a document e.g. blog entry, forum entry
<aside>	A piece of content that is somehow related to the rest of the page
<audio>	Audio media content
<canvas>	A component for rendering dynamic bitmap graphics on the fly. e.g games
<command>	A command that the user can invoke: a button, radio button or checkbox
<datalist>	Together with the new list attribute for the <input> element can be used to make combo boxes
<details>	Additional information or controls that the user can obtain on demand, to provide details on the document, or parts of it
<embed>	Used for plug-in content
<figure>	A piece of self-contained flow content referenced as a single unit from the main flow of the document
<figcaption>	Caption for a <figure>
<footer>	Footer for a section; may contain information about author, copyright information, etc.
<header>	A group of introductory or navigation aids
<hgroup>	Header of a section
<keygen>	A key pair generation control for user authentication in forms
<mark>	A run of text in one document marker or highlighted for reference purposes

“new shiny”

HTML



Cross-document
messaging

Multimedia

WebGL

Offline storage

Markup
improvements

Forms

Geolocation

CSS3

Document
editing

Microdata

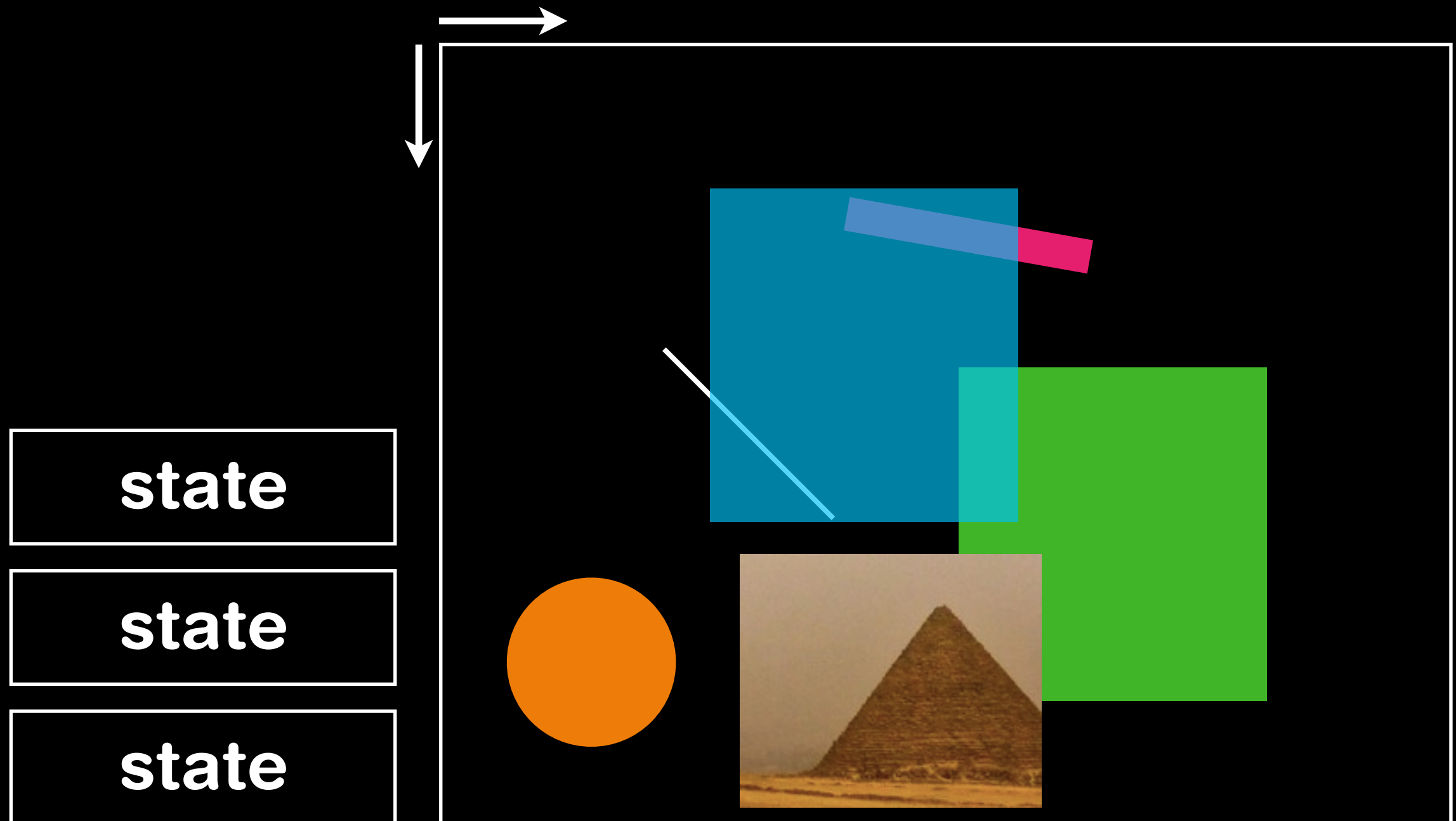
Canvas

History
management

File API

Drag-and-drop

Canvas



<http://vj.jole.fi/>

CONTENTS INCLUDE:

- Introduction to Canvas
- Browser Support and Hardware Acceleration
- What Canvas Can and Cannot Do
- A Comparison with SVG
- Canvas Performance
- Creating a Canvas and More!

HTML 5 Canvas

A Web Standard for Dynamic Graphics

By Simon Sarris

INTRODUCTION TO CANVAS

The HTML <canvas> element allows for on-the-fly creation of graphs, diagrams, games, and other visual elements and interactive media. It also allows for the rendering of 2D and 3D shapes and images, typically via JavaScript.

```
<canvas id="canvas1" width="500" height="500"></canvas>
<script type="text/javascript">
var can = document.getElementById('canvas1');
var ctx = can.getContext('2d');

ctx.fillText("Hello World!", 50, 50);
</script>
```

Canvas is perhaps the most visible part of the new HTML5 feature set, with new demos, projects, and proofs of concept appearing daily.

Canvas is a very low-level drawing surface with commands for making lines, curves, rectangles, gradients and clipping regions built in. There is very little else in the way of graphics drawing, which allows programmers to create their own methods for several basic drawing functions such as blurring, tweening, and animation. Even drawing a dotted line is something that must be done by the programmer from scratch.

Canvas is an immediate drawing surface and has no scene graph. This means that once an image or shape is drawn to it, neither the Canvas nor its drawing context have any knowledge of what was just drawn.

For instance, to draw a line and have it move around, you need to do much more than simply change the points of the line. You must clear the Canvas (or part of it) and redraw the line with the new points. This contrasts greatly with SVG, where you would simply give the line a new position and be done with it.



You can visit the evolving specification for Canvas at the WHATWG site: <http://www.whatwg.org/specs/web-apps/current-work/multipage/the-canvas-element.html>.

Browser Support and Hardware Acceleration

Canvas is supported by Firefox 1.5 and later; Opera 9 and later; and newer versions of Safari, Chrome, and Internet Explorer 9 and 10.

The latest versions of these browsers support nearly all abilities of the Canvas element. A notable exception is drawFocusRing, which no browser supports effects.

Hardware acceleration is supported in some variation by all current browsers, though the performance gains differ. It is difficult to benchmark between the modern browsers because they are changing frequently, but so far IE9 seems to consistently get the most out of having a good GPU. On a machine with a good video card it is almost always the fastest at rendering massive amounts of images or canvas-to-canvas draws.

Accelerated IE9 also renders fillRect more than twice as fast as the other major browsers, allowing for impressive 2D particle effects [1]. Chrome often has the fastest path rendering but can be inconsistent between releases. All browsers render images and rects much faster than paths or text, so it is best to use images and rects if you can regardless of which browsers you are targeting.

	Canvas	SVG
Support	<ul style="list-style-type: none"> • All modern versions of Chrome, Safari, Firefox, and Opera have at least some support. Internet Explorer 9+ has support. Almost all modern smart phones. • Internet Explorer 7 and 8 have limited support through the excanvas library. • Rapidly growing in popularity 	<ul style="list-style-type: none"> • SVG support in all modern browsers. Almost all modern smart phones.
Statefulness	<ul style="list-style-type: none"> • Bitmapped, immediate drawing surface • Shapes are drawn and nothing is remembered about their state. 	<ul style="list-style-type: none"> • Vector-based, retained drawing surface • Every drawn shape is a DOM object.
Other Considerations	<ul style="list-style-type: none"> • Generally faster • All event handling and statefulness must be programmed yourself. • Canvas will be effectively disabled (rendering nothing) if scripting is disabled. 	<ul style="list-style-type: none"> • Generally slower, especially past 10,000 objects. • Since all SVG elements are DOM objects, statefulness is built in and event handling is much easier. • Easier for a designer to work with, many programs such as Illustrator can output SVG • SVG has built-in support for animation.
Accessibility	<ul style="list-style-type: none"> • Difficult to interface with other DOM objects • Working with text can be difficult. • Recreating text-based DOM element functionality is strongly advised against, even in the specification itself. • Cannot operate when scripting is disabled. 	<ul style="list-style-type: none"> • All SVG objects are already DOM objects. • Text is searchable by the browser and web crawlers.



Windows® Internet
Explorer 9

//Enhance your site
//with pinning today
//It takes less than an hour

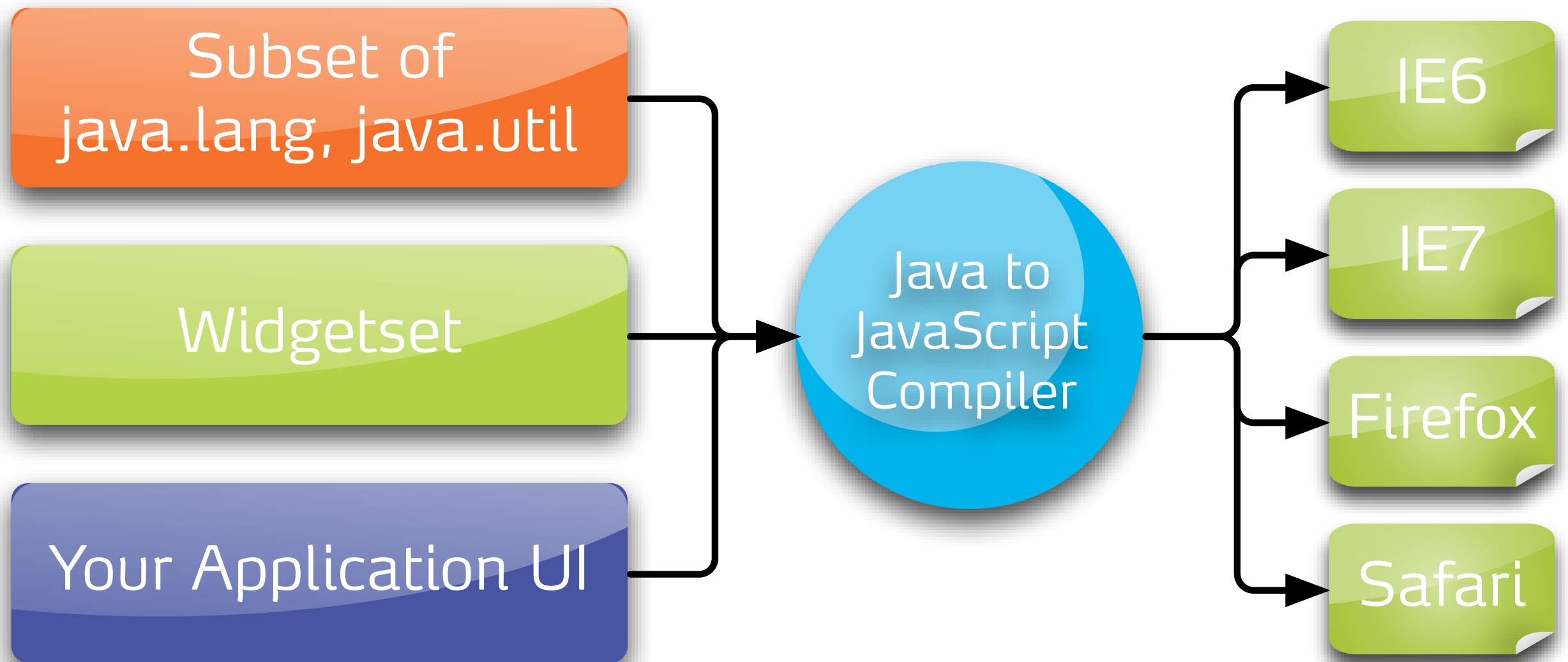
buildmypinnedsite.com

Get the code

Microsoft



Google Web Toolkit



simpler

- 100% Java
- Static typing
- Object oriented
- Excellent tooling

less bugs

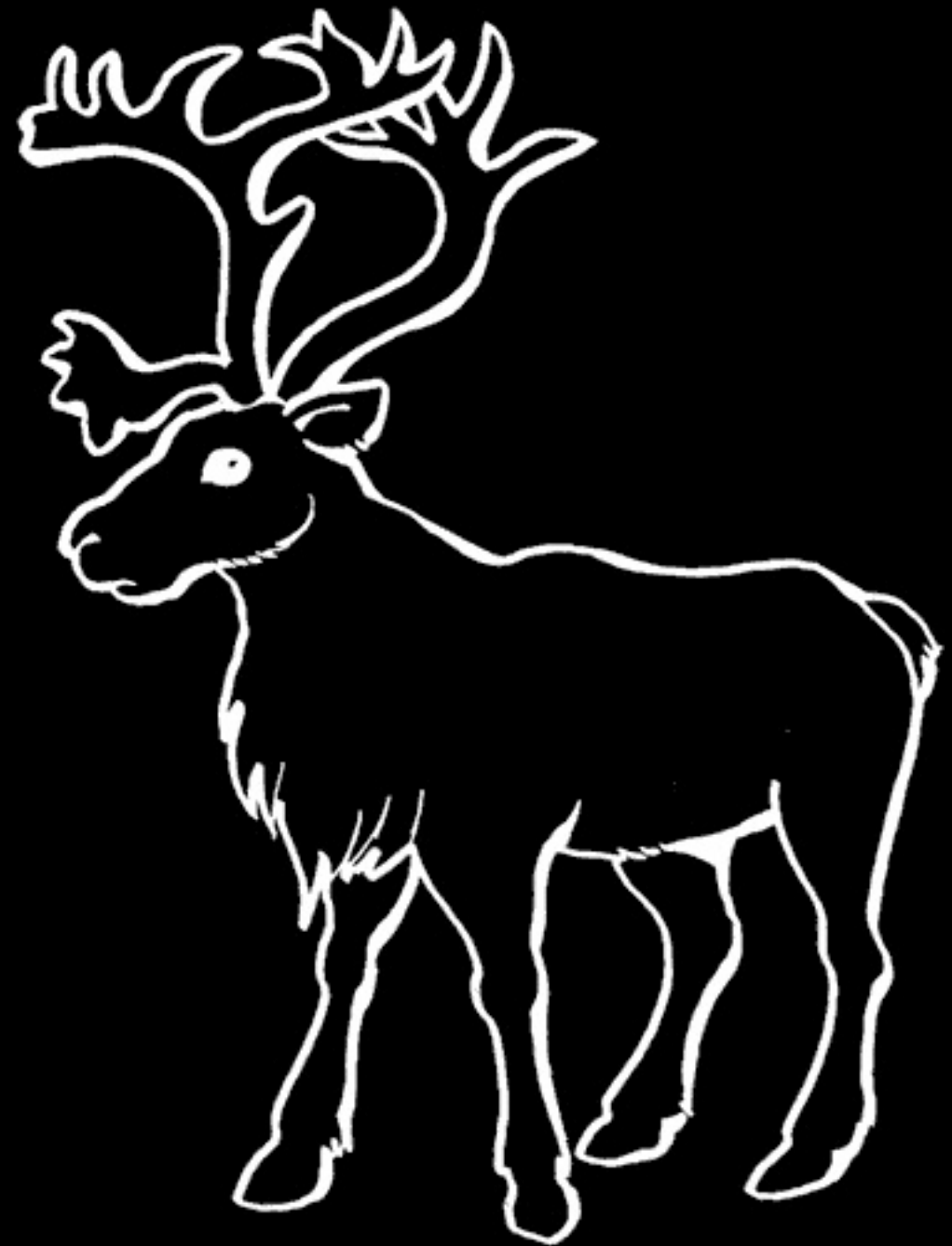
- Stop debugging JavaScript spaghetti
- Ignore most browser differences

client-side

- UI in client
- Asynchronous RPC
- Services (for UI)

vaadin }>

vaadin }>



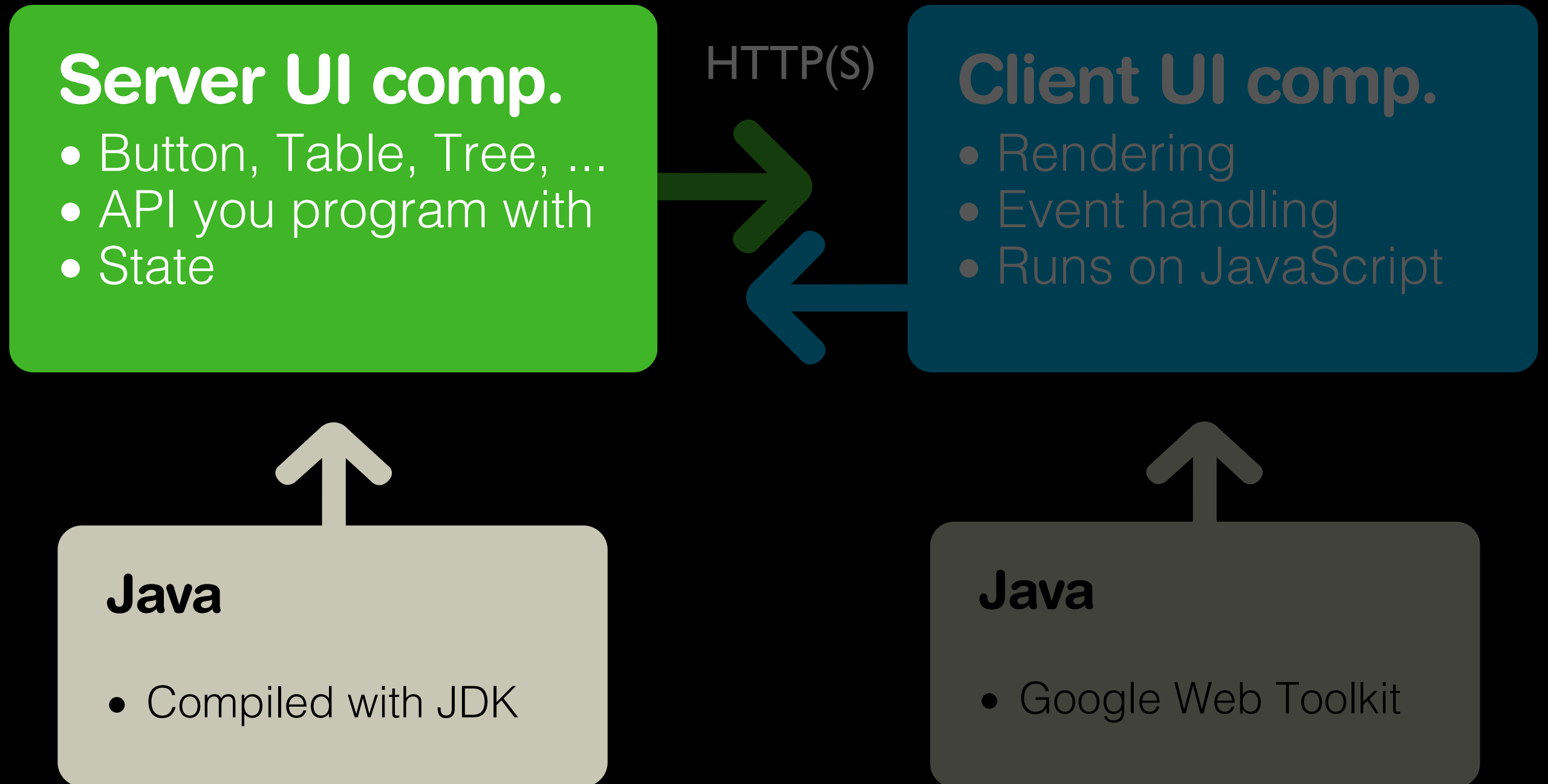
**Vaadin is a
UI framework
for rich web
applications**

java } html >

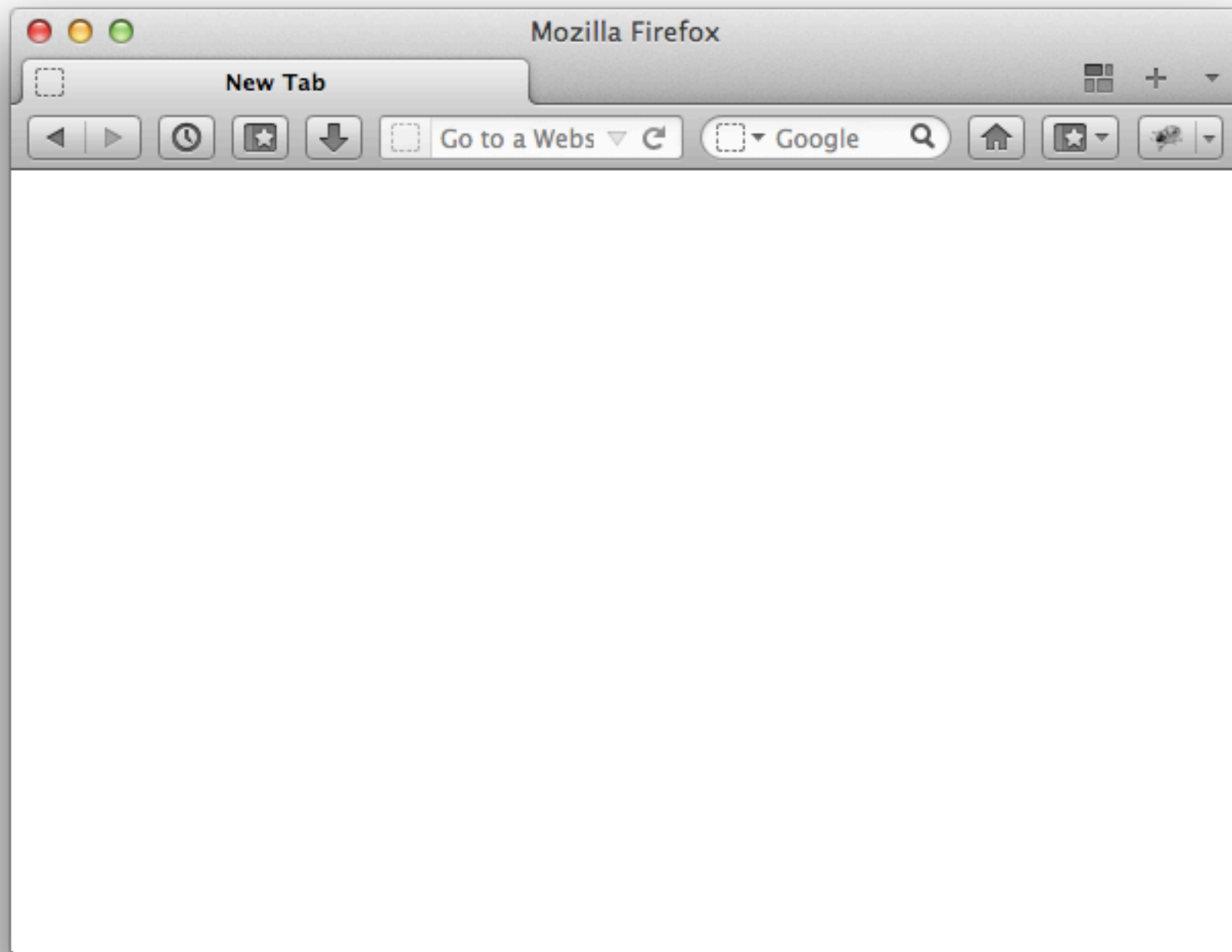
Web application layers

	Backend server	Web server	RPC	Java to JavaScript	JavaScript
Vaadin	required	required	optional	optional	optional
GWT	required	required	required	required	optional
JS	required	required	required		required

Vaadin UI component architecture

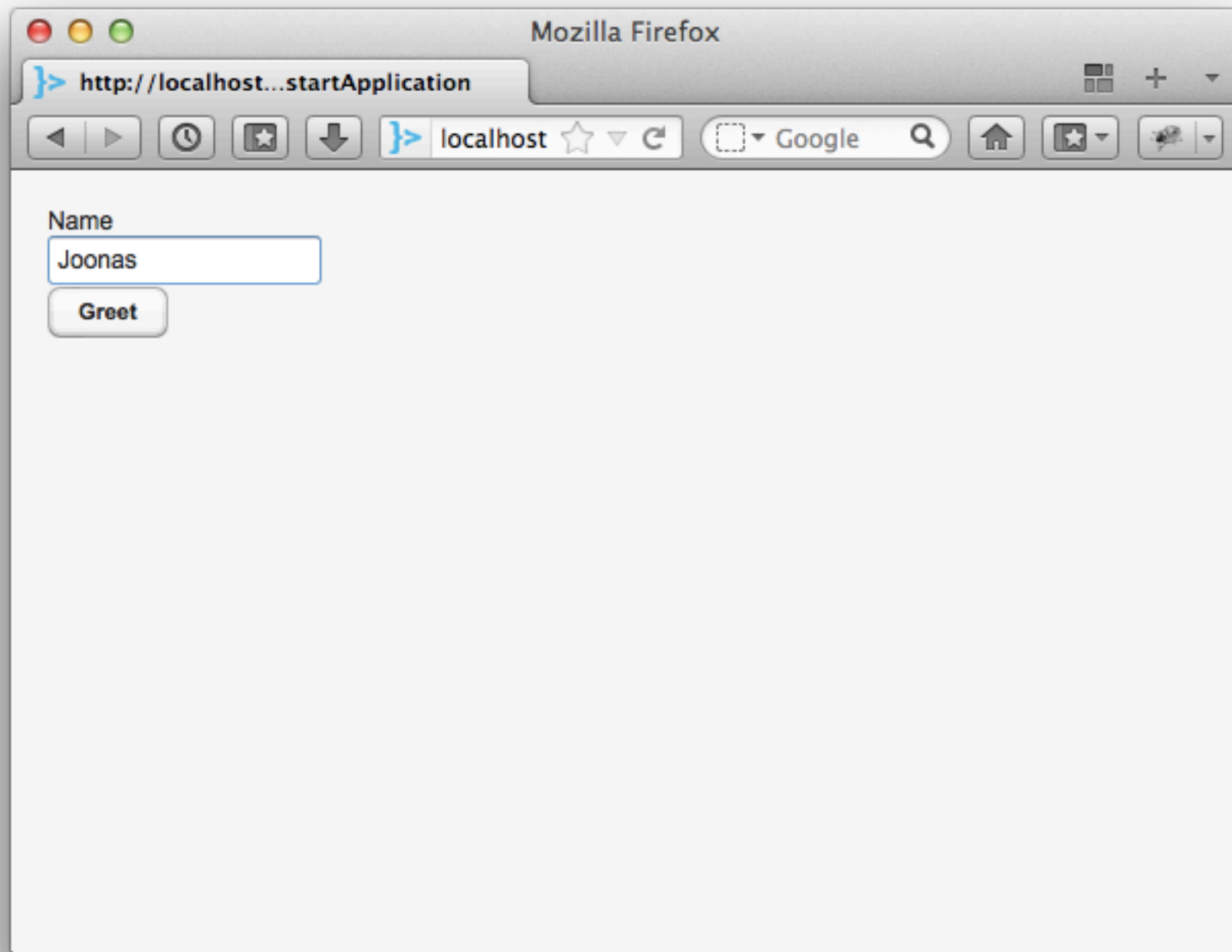


How does it work, really?



- Initial HTML
- CSS (theme)
- Images
- JavaScript

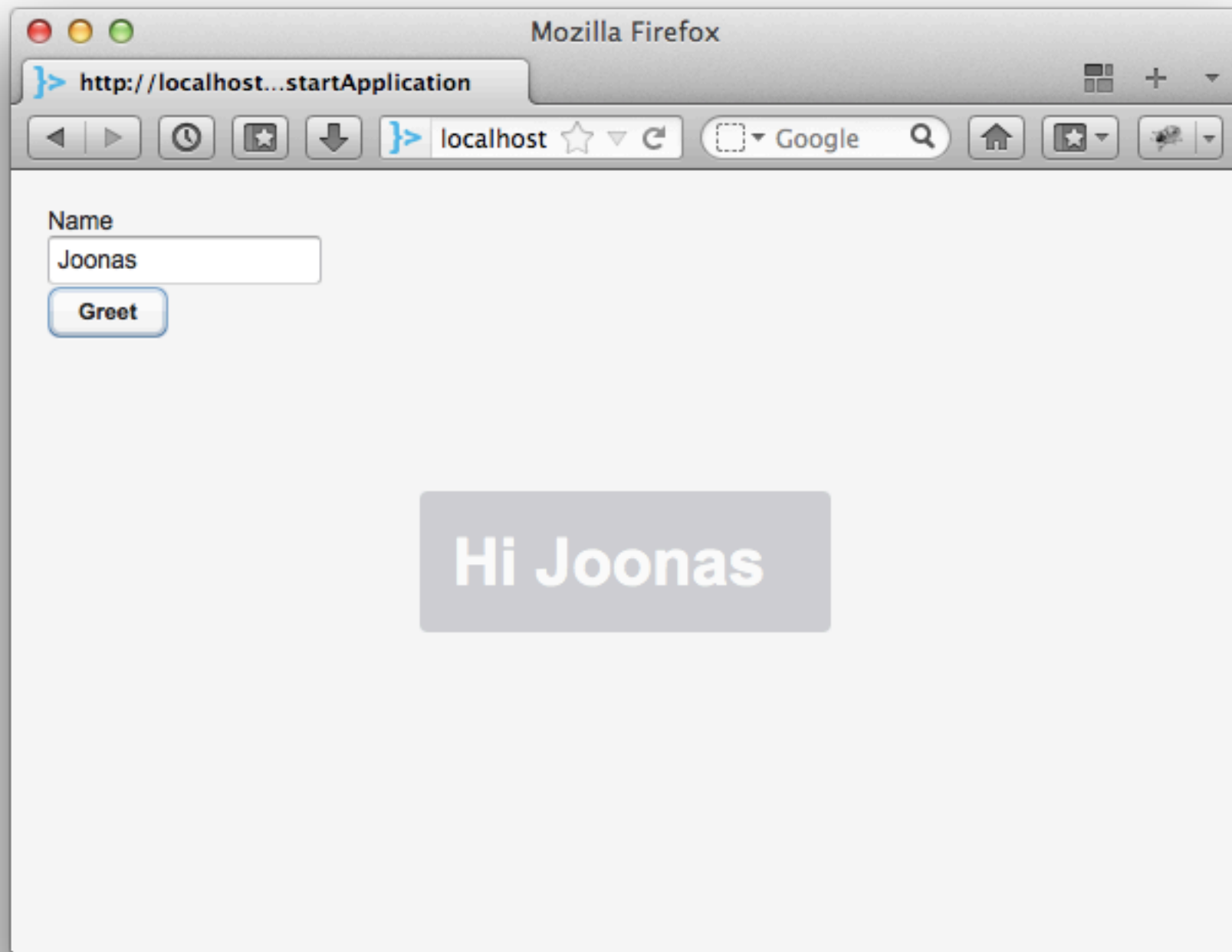
250k total




- name="Joonas"
- button clicked

150 bytes

```
public void buttonClick(ClickEvent event) {  
    mainWindow.showNotification("Hi " + name);  
}
```



- 
- name="Joonas"
 - button clicked

150 bytes

- 
- Add notification

466 bytes



2

DASHBOARD



SALES



TRANSACTIONS

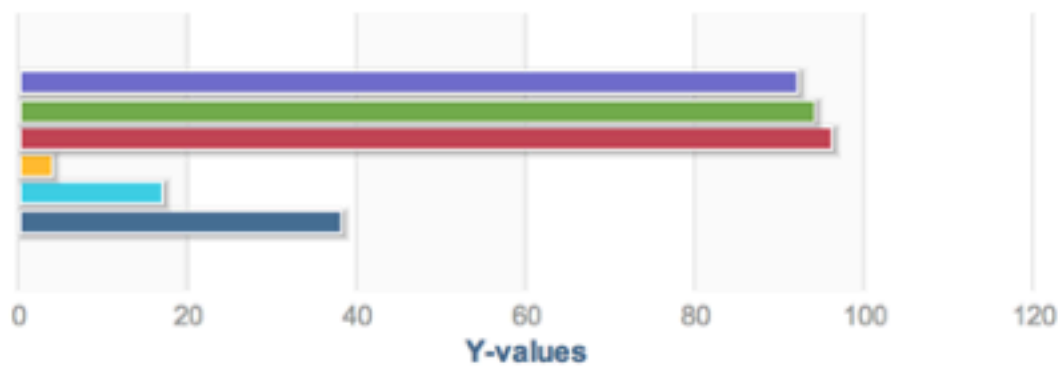


REPORTS



SCHEDULE

TOP GROSSING MOVIES



Parker Hansel and Gretel: Witch Hunters Movie 43
Monsters, Inc. 3D Zero Dark Thirty Silver Linings Playbook

NOTES



Remember to:

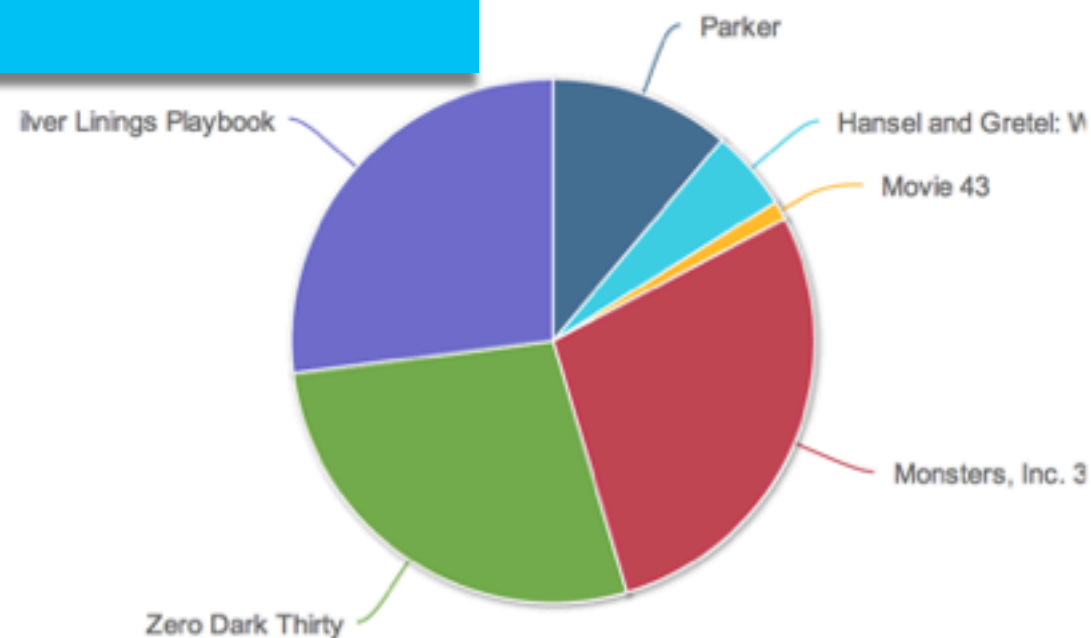
- Zoom in and out in the Sales view
- Filter the transactions and drag a set of them to the Reports tab
- Create a new report
- Change the schedule of the movie theater

[http://demo.vaadin.com/
dashboard](http://demo.vaadin.com/dashboard)

TOP 10 TITLES BY REV

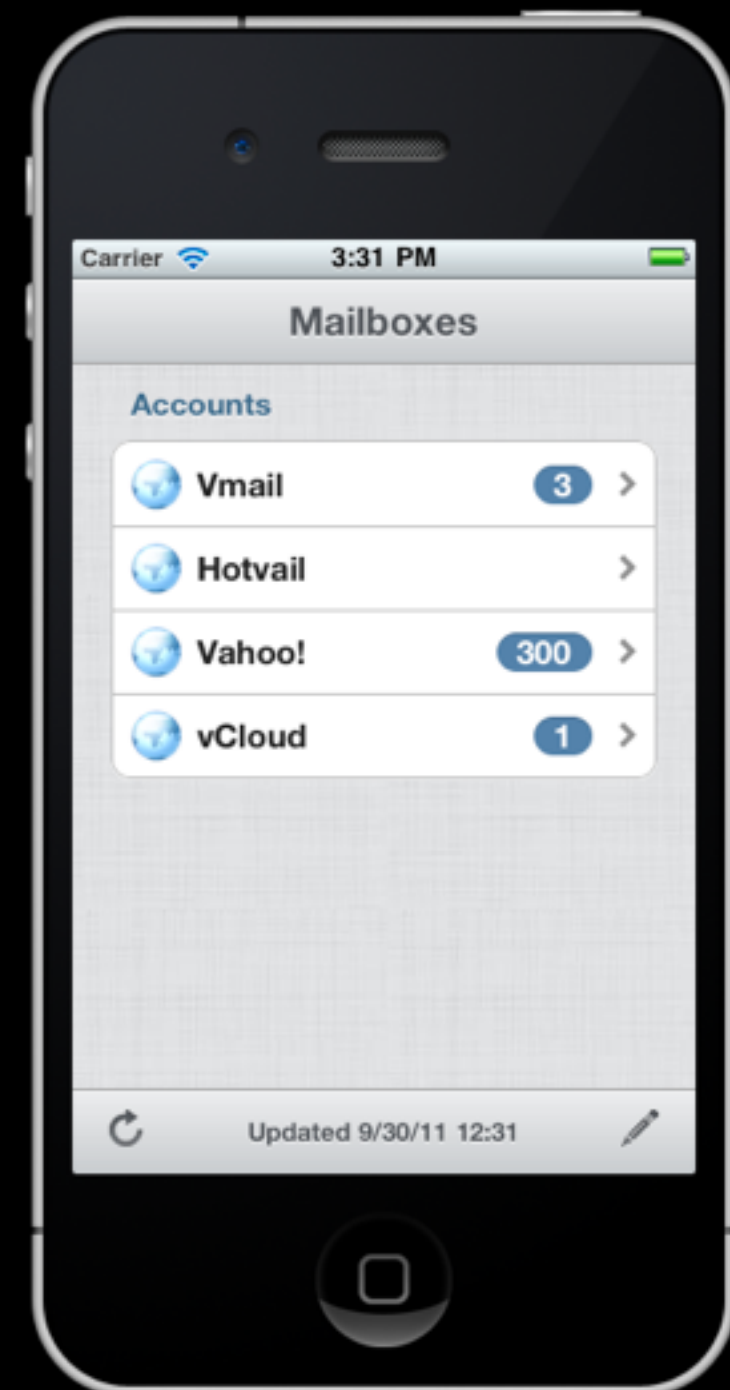
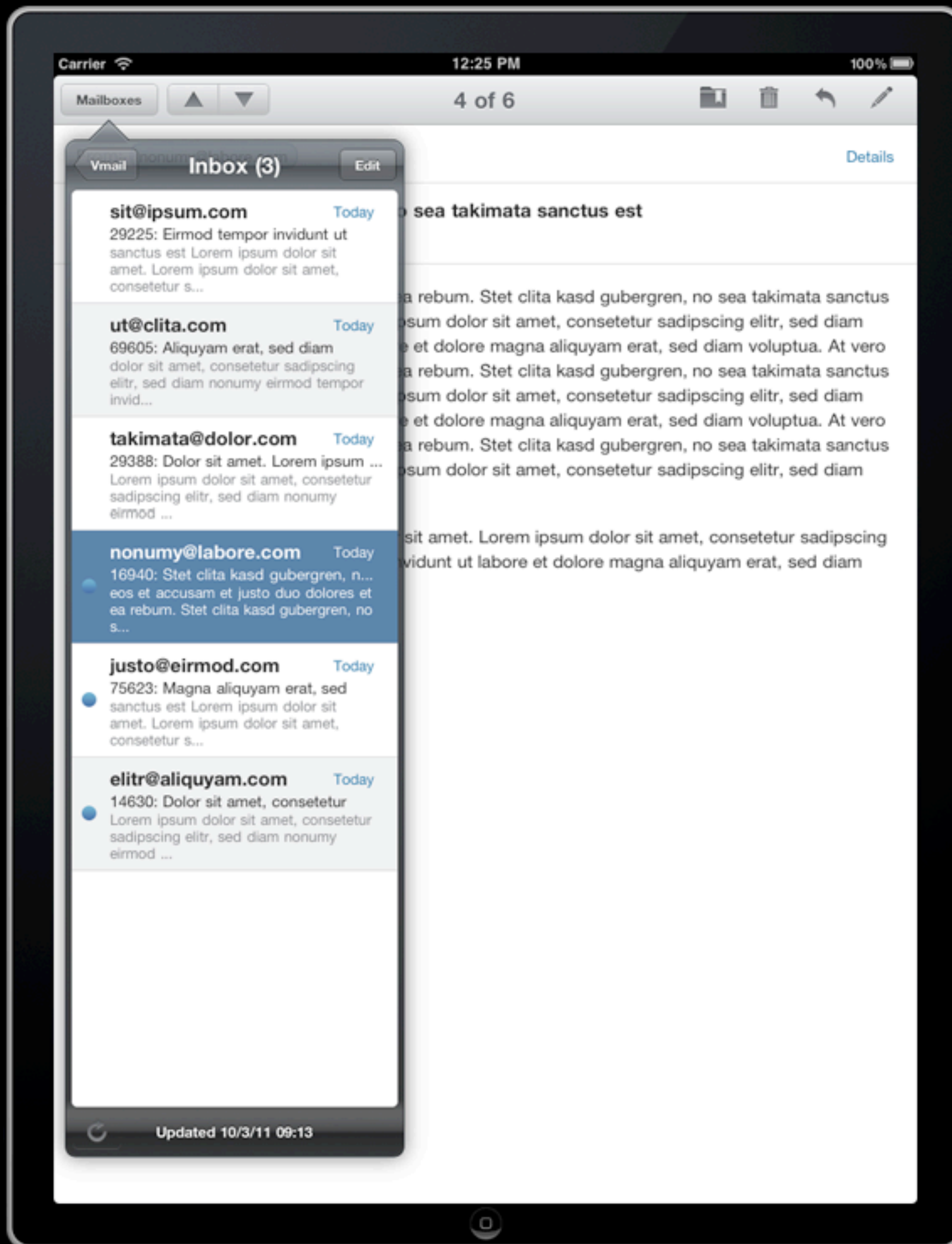


	Title	
1	Skyfall	\$603021.03
2	Wreck-it Ralph	\$56978.32
3	Lincoln	\$55928.02
4	Silver Linings Playbook	\$54418.14
5	Life of Pi	\$50882.59
6	Flight	\$48055.12
7	Monsters, Inc. 3D	\$41058.53
8	Rise of the Guardians	\$34406.37
9	Django Unchained	\$34352.09
10	Zero Dark Thirty	\$29014.34



Vaadin Charts

Dave
Johnson



Directory

All versions

Browse

All

[UI Components](#)
[Data Components](#)
[Themes](#)
[Tools](#)
[Miscellaneous](#)
[Official](#)
[Guest](#)
[Authoring](#)
[Subscribe RSS](#)
[Help](#)
[FAQ](#)
[Feedback](#)
[Most Recent](#) [Highest Rated](#) [Top Downloads](#)

Showing [CERTIFIED](#) [STABLE](#) [BETA](#) [EXPERIMENTAL](#)

« Previous [Next »](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [11](#) [12](#) ... [27](#)

316 Results

CSVValidation

In [UI Components](#) by [Marko Grönroos](#)

Client-side validation of text fields

Version 0.5.1 BETA

★★★★★ 3

↓ 1220

SimpleCalendar

In [UI Components](#) by [John Ahlroos](#)

Date selection made easy.

Version 0.1 EXPERIMENTAL No ratings yet

↓ 2

PopupButton

In [UI Components](#) by [Henri Kerola](#)

Button with popup

Version 2.2.0 BETA

★★★★★ 19

↓ 8546

Number Field

In [UI Components](#) by [Francesco Portus](#)

Numeric text field that allows insert only a valid numbers with keyboard and also allows spin value up/down.

Version 1.0.2 BETA

★★★★★ 2

↓ 99

ExternalLayout

In [UI Components](#) by [Risto Yrjänä](#)

ExternalLayout allows Vaadin components to be rendered outside the normal Vaadin application DOM-hierarchy

Version 2.0.1 STABLE

★★★★★ 1

↓ 186

FilterableTwinColSelect

In [UI Components](#) by [Kim Leppänen](#)

An improved version of the core's TwinColSelect, allows you to filter values

Version 1.0.0 EXPERIMENTAL No ratings yet

↓ 1

MockupContainer

In [Data Components](#) by [Risto Yrjänä](#)

Jain I18N

In [Data Components](#) by [Lokesh Jain](#)

Download for Free
vaadin.com/book

The background features a large, stylized Vaadin logo in a light beige color. The logo is composed of several thick, rounded lines that form a shape reminiscent of a stylized 'V' or a compass rose. It is positioned on the right side of the image, partially overlapping the white background and the dark grey background.

Book of Vaadin

~700 pages



Designing Step-by-step



Designing Step-by-step

Plan



Prototype



Design



Implement

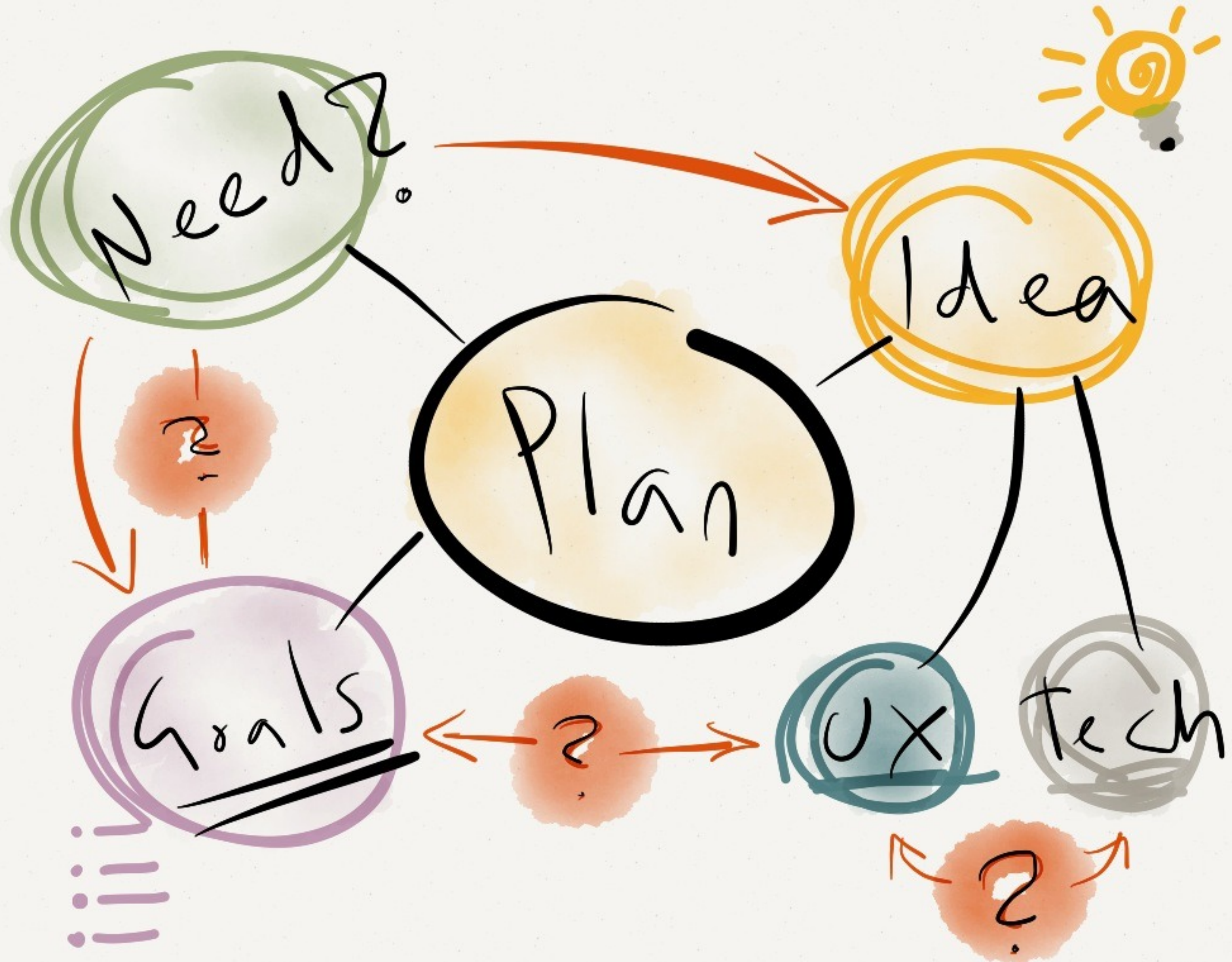


Test



Distribute





Need

We can not get
the **UX** we need
with the **existing
widgets**

Goals

- List of
- real
- quantifiable
- requirements
- for UX

Example goals

- Load and view data in XLS files
- Show visual overview for numeric cols
- Must support 1000 cell tables
- Supports the latest Firefox & Chrome

**Nail down
the minimum viable
set of supported
browser versions
with the customer**

Idea

=

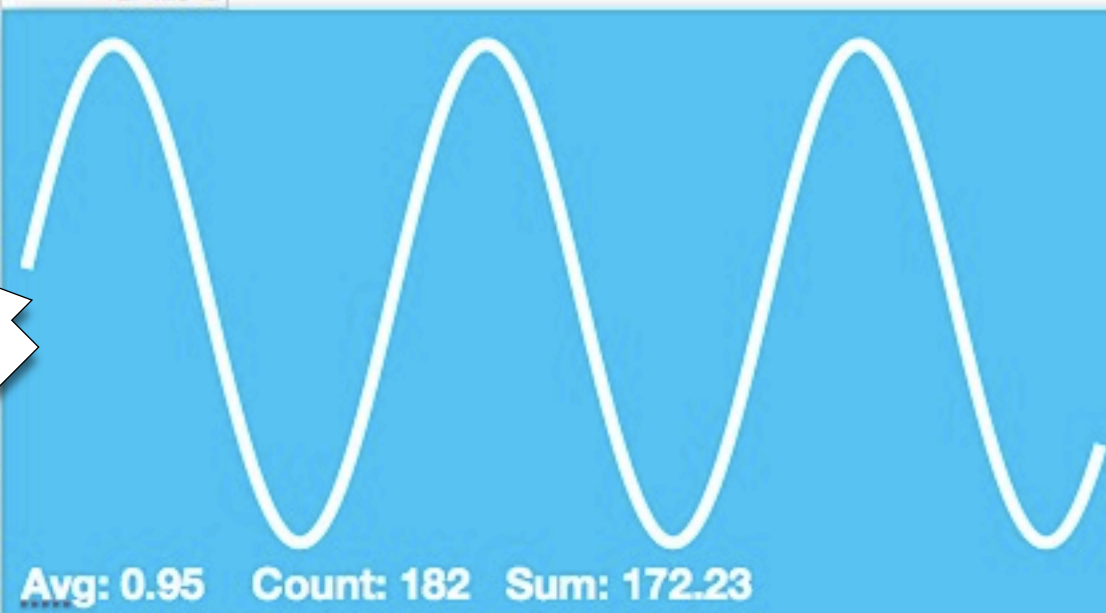
UX [how it is used]

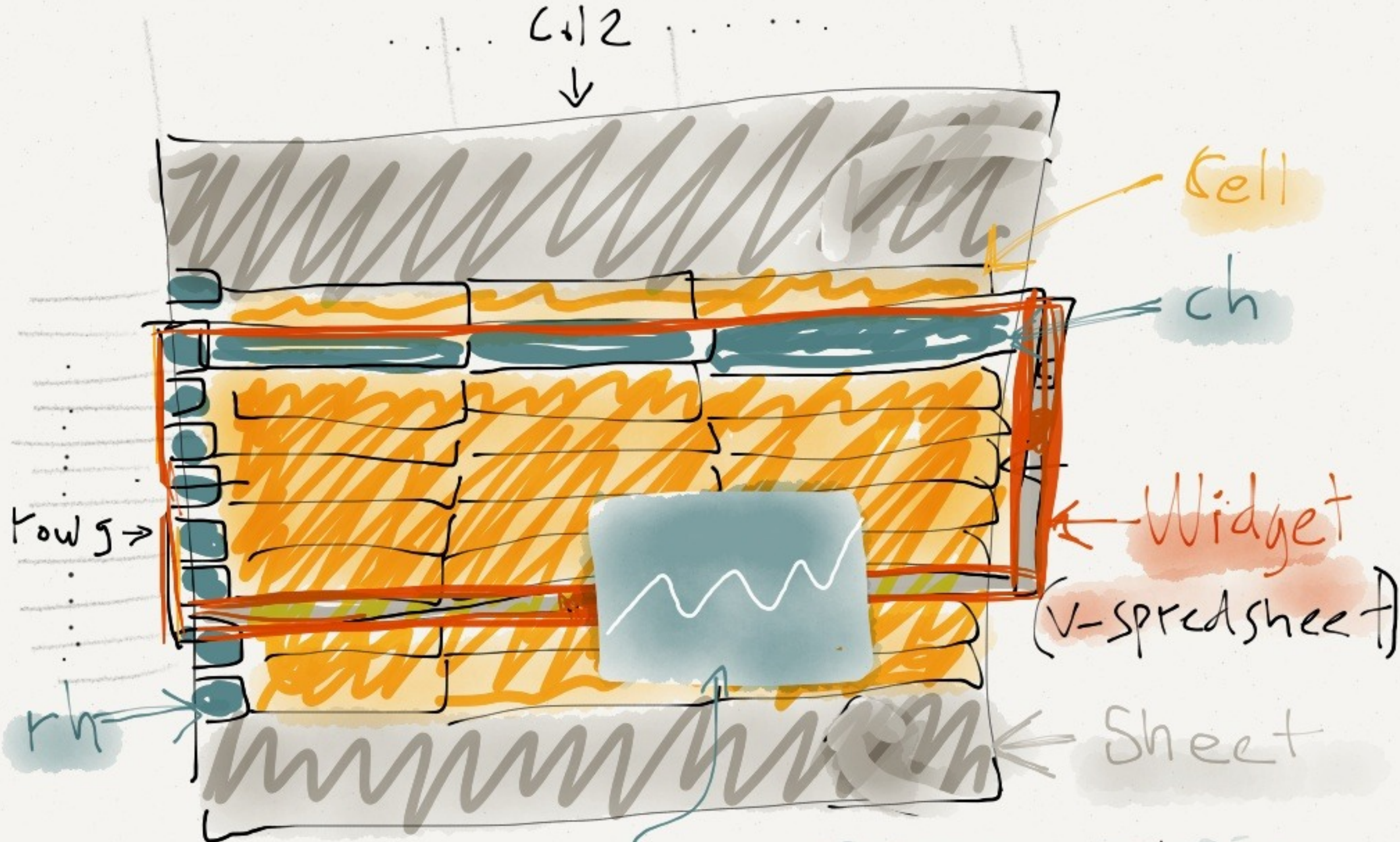
+

Tech [how it works]

Sample	Measured value	Date
1	9,983341664683	1.1.2012
2	19,86693307951	2.1.2012
3	29,55202066613	3.1.2012
4	38,94183423087	4.1.2012
5	47,94255386042	5.1.2012
6	56,4642473395	6.1.2012
7	64,42176872377	7.1.2012
8	71,73560908995	8.1.2012
9	78,33269096275	9.1.2012
10	84,14709848079	10.1.2012
11	89,12073600614	11.1.2012
12	93,20390859672	12.1.2012
13	96,35581854172	13.1.2012
14	98,54497299885	14.1.2012
15	99,74949866041	15.1.2012
16	99,95736030415	16.1.2012
17	99,16648104525	17.1.2012
18	97,38476308782	18.1.2012
19	94,63000876874	19.1.2012
20	90,92974268257	20.1.2012
21	86,32093666489	21.1.2012
22	80,84964038196	22.1.2012
23	74,57052121767	23.1.2012
24	67,54631805512	24.1.2012
25	59,8472144104	25.1.2012
26	51,55013718215	26.1.2012
27	42,73798802338	27.1.2012
28	33,49881501559	28.1.2012
29	23,9249329214	29.1.2012
30	14,11200080599	30.1.2012
31	4,158066243329	31.1.2012
32	-5,837414342758	1.2.2012
33	-15,77456941432	2.2.2012
34	-25,55411020268	3.2.2012
35	-35,07832276896	4.2.2012
36	-44,25204432949	5.2.2012
37	-52,98361409085	6.2.2012
38	-61,18578909427	7.2.2012
39	-68,7766159184	8.2.2012
40	-75,68024953079	9.2.2012
41	-81,82771110644	10.2.2012
42	-87,15757724136	11.2.2012
43	-91,61659367495	12.2.2012
44	-95,16020738895	13.2.2012
45	-97,75301176651	14.2.2012
46	-99,36910036335	15.2.2012
47	-99,99232575641	16.2.2012

Sample	Measured value	Date
1	9,983341664683	1.1.2012
2	19,86693307951	2.1.2012
3	29,55202066613	3.1.2012
4	38,94183423087	4.1.2012
5	47,94255386042	5.1.2012
6	56,4642473395	6.1.2012
7	64,42176872377	7.1.2012
8	71,73560908995	8.1.2012
9	78,33269096275	9.1.2012
10	84,14709848079	10.1.2012
11	89,12073600614	11.1.2012
12	93,20390859672	12.1.2012
13	96,35581854172	
14	98,54497299885	
15	99,74949866041	
16	99,95736030415	
17	99,16648104525	
18	97,38476308782	
19	94,63000976874	
20	90,929742	
21	86,3209366	
22	80,84964038	
23	74,57052121767	
24	67,54631805512	
25	59,8472144104	
26	51,55013718215	
27	42,73798802338	
28	33,49881501559	
29	23,9249329214	29.1.2012
30	14,11200080599	30.1.2012
31	4,158066243329	31.1.2012
32	-5,837414342758	1.2.2012
33	-15,77456941432	2.2.2012
34	-25,55411020268	3.2.2012
35	-35,07832276896	4.2.2012
36	-44,25204432949	5.2.2012
37	-52,98361409085	6.2.2012
38	-61,18578909427	7.2.2012
39	-68,7766159184	8.2.2012
40	-75,68024953079	9.2.2012
41	-81,82771110644	10.2.2012
42	-87,15757724136	11.2.2012
43	-91,61659367495	12.2.2012
44	-95,16020738895	13.2.2012
45	-97,75301176651	14.2.2012
46	-99,36910036335	15.2.2012
47	-99,99232575641	16.2.2012





Dom & styles

Graph
(canvas)

SpreadsheetView

**Always start from
defined goals -
never let idea to
rule your design**

4 Performance



Proto

1 Static HTML



2 Key UX in JS



3 Cross Browser Compatibility

```

.v-spreadsheet .row13 { height: 12px; top: 177px;}
.v-spreadsheet .row14 { height: 12px; top: 190px;}
.v-spreadsheet .col1 { width: 99px; left: 50px;}
.v-spreadsheet .col2 { width: 99px; left: 150px;}
.v-spreadsheet .col3 { width: 99px; left: 250px; }
.v-spreadsheet .col4 { width: 99px; left: 350px; }
.v-spreadsheet .col5 { width: 99px; left: 450px; }
.v-spreadsheet .col6 { width: 99px; left: 550px; }
</style>
</head>
<body>

<div class="v-spreadsheet" id="elem">
  <div class="sheet" id="sheet">

    <div class="c col1 row1"></div>
    <div class="c col2 row1"></div>
    <div class="c col3 row1"></div>
    <div class="c col4 row1"></div>
    <div class="c col5 row1"></div>
    <div class="c col6 row1"></div>
    <div class="rh row1">1</div>

    <div class="c col1 row2"></div>
    <div class="c col2 row2"></div>
    <div class="c col3 row2"></div>
    <div class="c col4 row2"></div>
    <div class="c col5 row2"></div>
    <div class="c col6 row2"></div>
    <div class="rh row2">2</div>

    <div class="c col1 row3"></div>

```



	A	B	C
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

**Works with
target browsers?**

padding-top: 2px;
border-right: 2px solid #d6d6d6;
border-bottom: 2px solid #d6d6d6;
border-top: 1px solid #d6d6d6;
}

</style>
</head>
<body>

<div class="v-spreadsheet" id="elem">
 <div class="sheet" id="sheet">
 </div>
</div>

<script>

```
var totalrows = 100;  
var totalcols = 13;  
var isIE = navigator&&navigator.userAgent&&navigator.userAgent.match(/bMSIE ([678])\./);  
function insertCSS(css) {  
  var stylesheet = document.styleSheets[0];  
  if (isIE) {  
    stylesheet.cssText += css;  
  } else {  
    stylesheet.insertRule(css, stylesheet.cssRules.length);  
  }  
}  
document.getElementById("elem").onscroll = function() {  
  var e = document.getElementById("elem");  
  var l = e.scrollLeft;  
  var classes = document.styleSheets[0].rules || document.styleSheets[0].cssRules
```

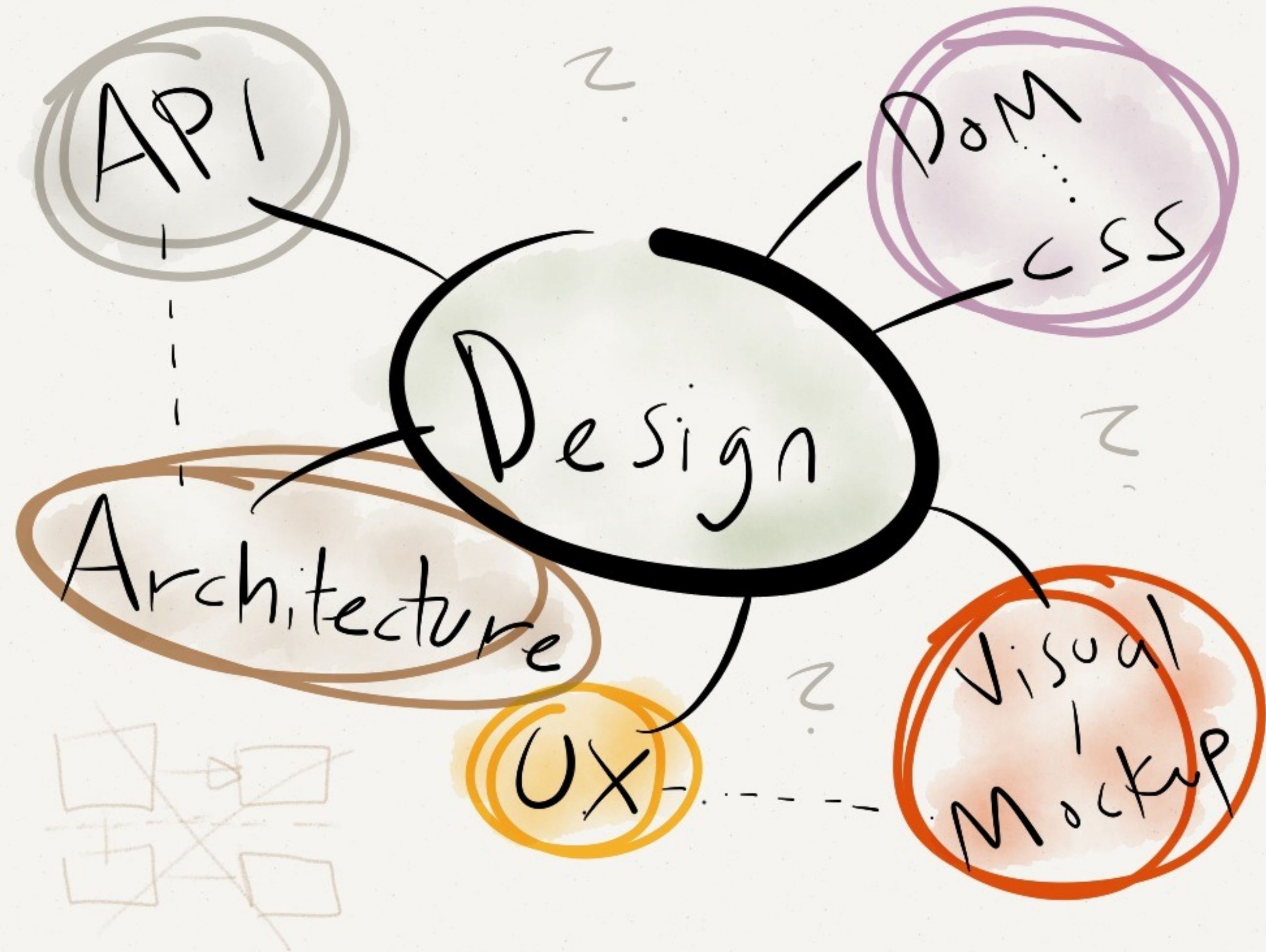
**Interactions
work with target
browsers?**

**Performance is
good enough for
target data?**

Proto DOD

- Includes main use-cases
- Works in target browsers
- Handles enough data

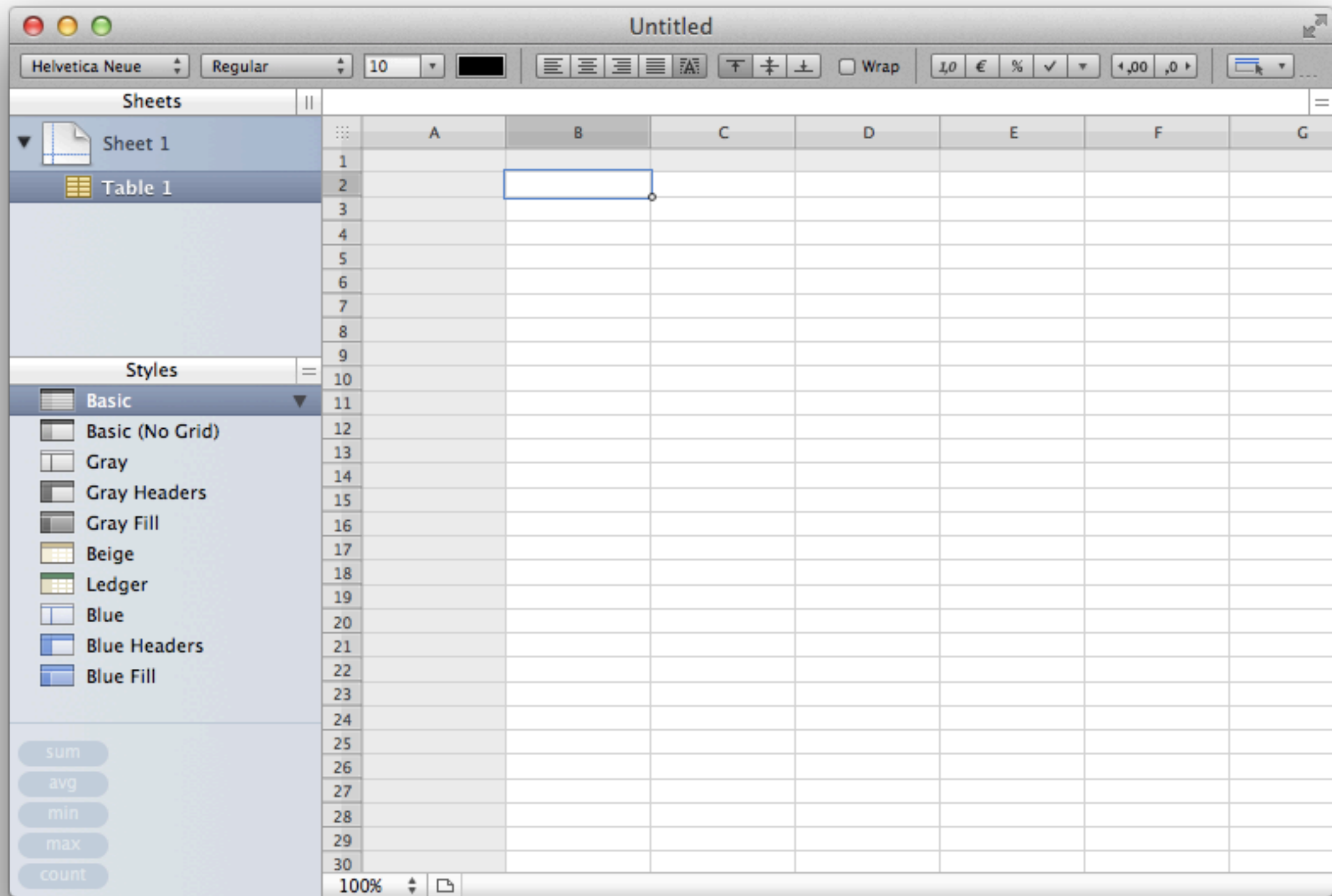
**Never start design
or implementation
before prototyping
browser compatibility
and performance**



**Design UX and
API first.**

**Never continue
from
the prototype
implementation**

**Drawing detailed
wireframes & mockups
and testing them with
users will save
time later**



**Shamelessly copy UX.
Then your users
already
know how to use it.**

WIDGET

BROWSER

Spreadsheet View

GWT

ABSTRACT COMPONENT CONNECTOR

Spreadsheet Model

COMPONENT
STAR

Spreadsheet Connector

Spreadsheet State

Vaadin

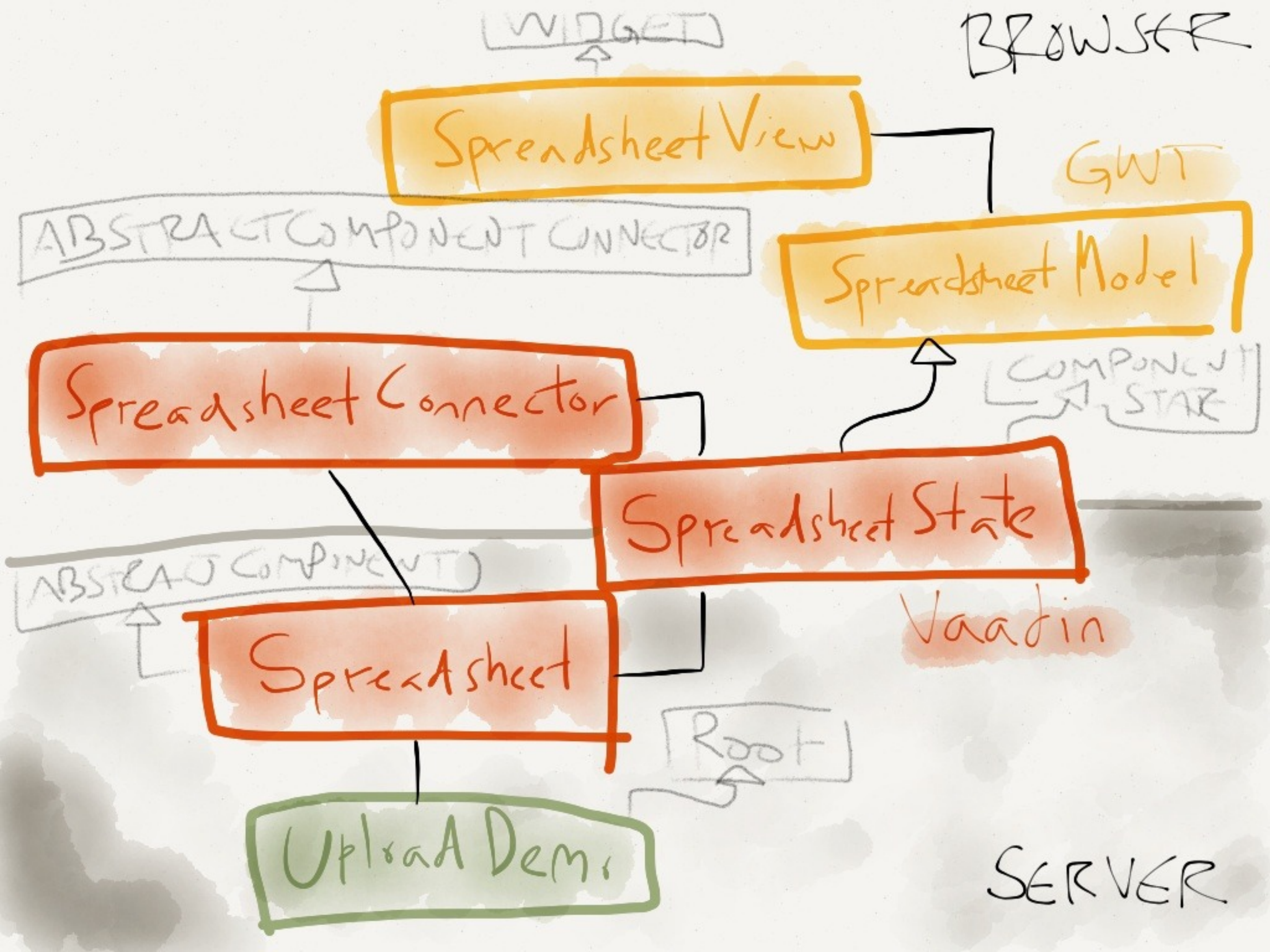
ABSTRACT COMPONENT

Spreadsheet

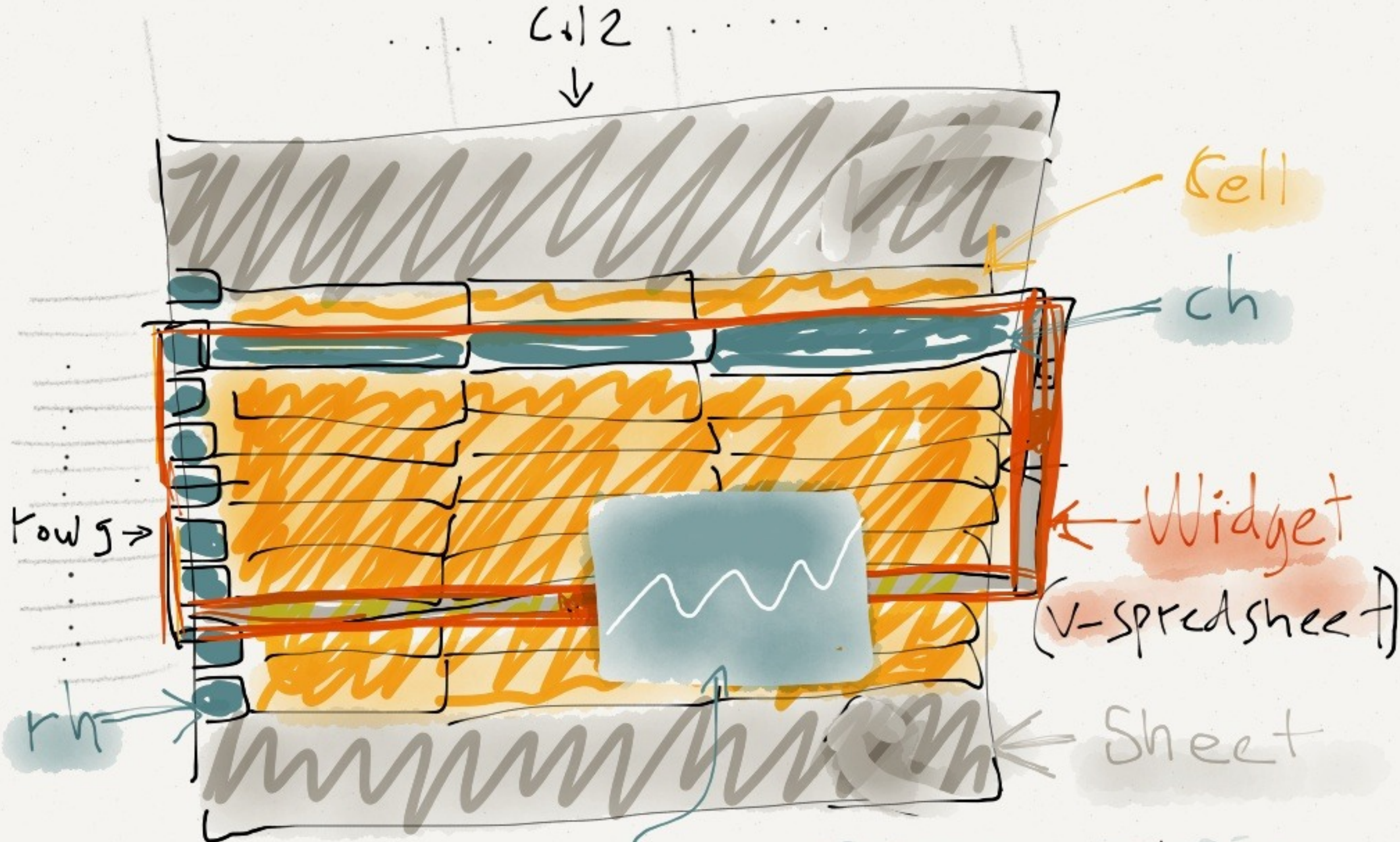
Root

Upload Demo

SERVER



**Aim for multilayered
design that lets your
users (developers)
change behavior of
your component**

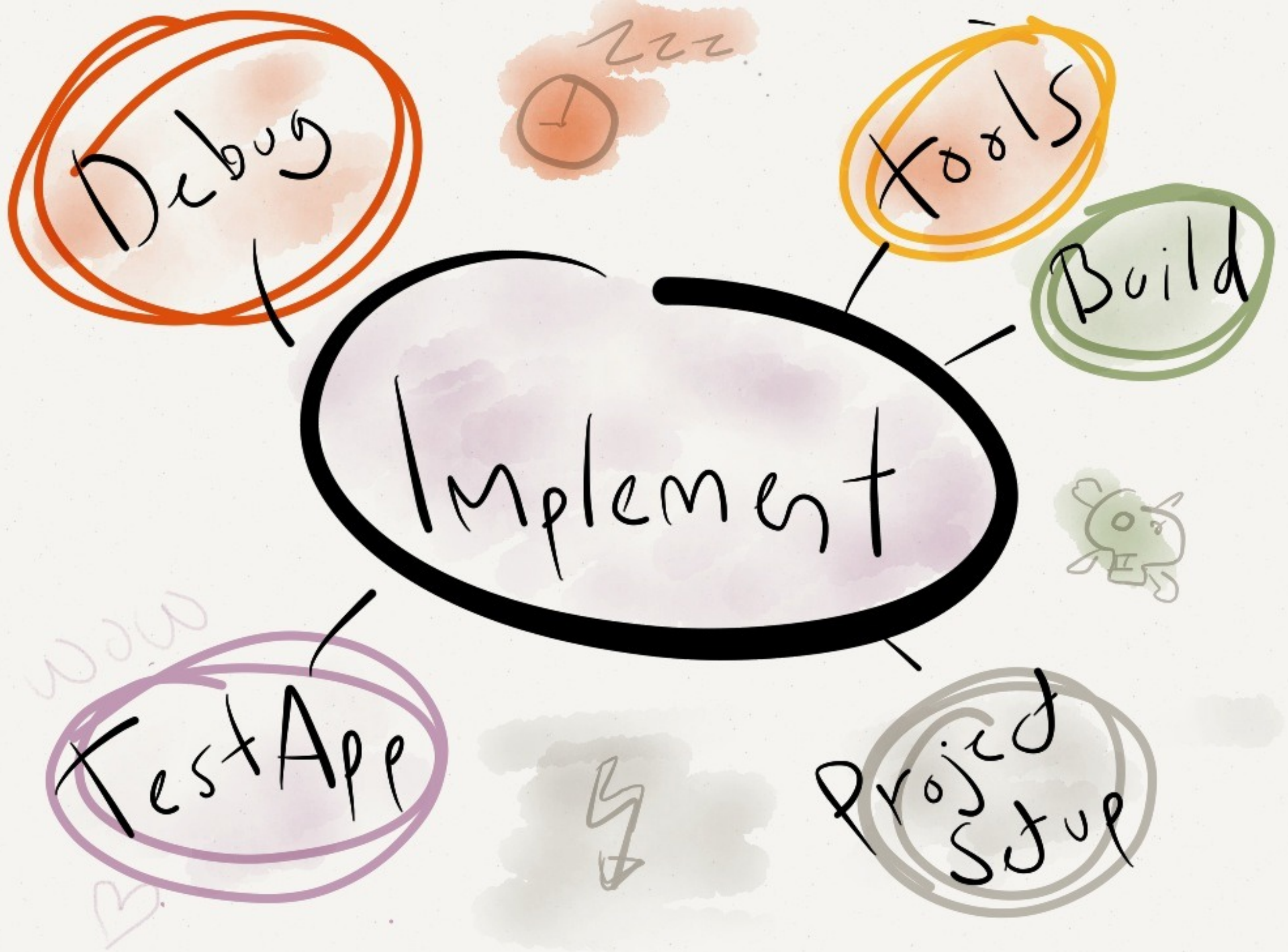


Dom & styles

Graph
(canvas)

























SpreadsheetView

**DOM classes and
CSS restrictions
must be documented
to make styling easy**



**Keep component
project separate
from your real
application project**

**Demo application
must include all
features
and serve as example
for your users**

- ▼  spreadsheet [spreadsheet master]
 - ▼  src/main/java
 - ▼  org.vaadin.spreadsheet
 - ▶  Spreadsheet.java
 - ▼  org.vaadin.spreadsheet.gwt
 - ▼  public
 - ▼  spreadsheet
 -  styles.css
 -  SpreadsheetWidgetSet.gwt.xml
 - ▼  org.vaadin.spreadsheet.gwt.client
 - ▶  SpreadsheetConnector.java
 - ▶  SpreadsheetModel.java
 - ▶  SpreadsheetState.java
 - ▶  SpreadsheetView.java
 -  rebel.xml
 - ▶  src/test/java
 - ▶  JRE System Library [J2SE-1.5]
 - ▶  Maven Dependencies
 - ▶  design
 - ▶  src
 - ▶  target
 -  licensing.txt
 -  pom.xml
 -  README.markdown

Invest in project
setup with a rapid
save-to-see cycle and a
robust build script.
These might not be the
same thing.

Component Add-on Project Setup HOWTO

This howto walks you through a complete setup for a project for developing, building and publishing your own Vaadin UI component add-ons. The goal here is not to teach how to write an add-on, but to make the process of setting up your project environment as smooth as possible. I hope this encourages you to try building and publishing your own add-ons :)

Goals for the project environment

- Fully automated build with Maven
- Allow anyone to re-build your project easily regardless of the IDE:s
- Almost instant save-build-deploy-try cycle
- Simple
- Project
- Easy

<https://vaadin.com/wiki/-/wiki/Main/Component+Add-on+Project+Setup+HOWTO>

Install

If you do not

- Eclipse IDE for Java EE developers from <http://www.eclipse.org> (Indigo Service Release 1 was used in this howto)
- Google Chrome browser from <https://www.google.com/chrome/> (other browsers will do, but Chrome is recommended)
- Eclipse plugins: m4e-wtp, vaadin, egit (optional) and jrebel (optional) from Marketplace (just select Help->Marketplace... from the menu)

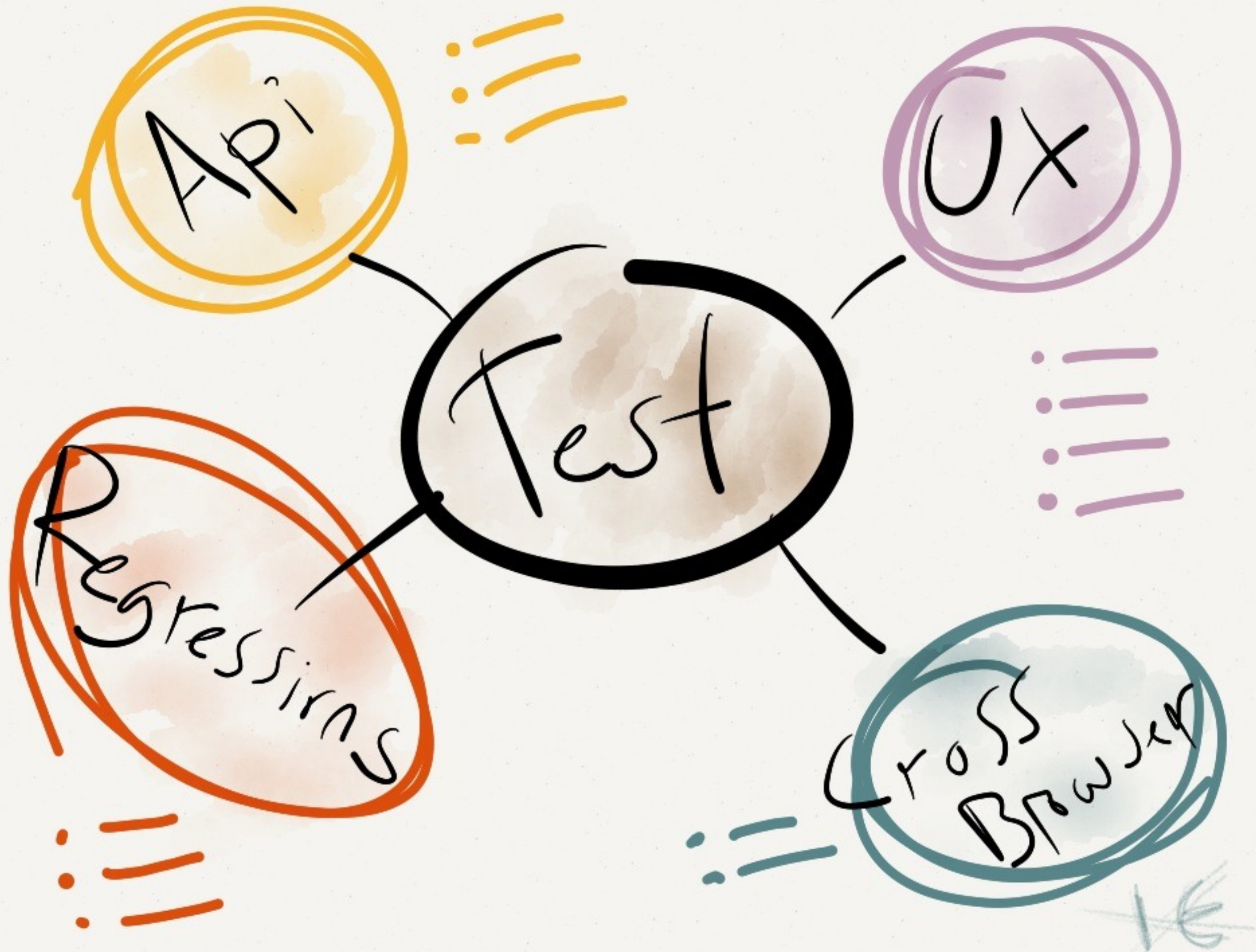
Create a new widget project

Start project creation wizard: File -> New -> Other... -> "Maven Project"

Give a proper name for your project and save it under workspace. For this example I am building a list widget and name it MyList.

Ensure that your Maven archetype catalogs contain <http://repo1.maven.org/maven2/archetype-catalog.xml> as remote catalog and select it.

Select vaadin-archetype-widget from the list.



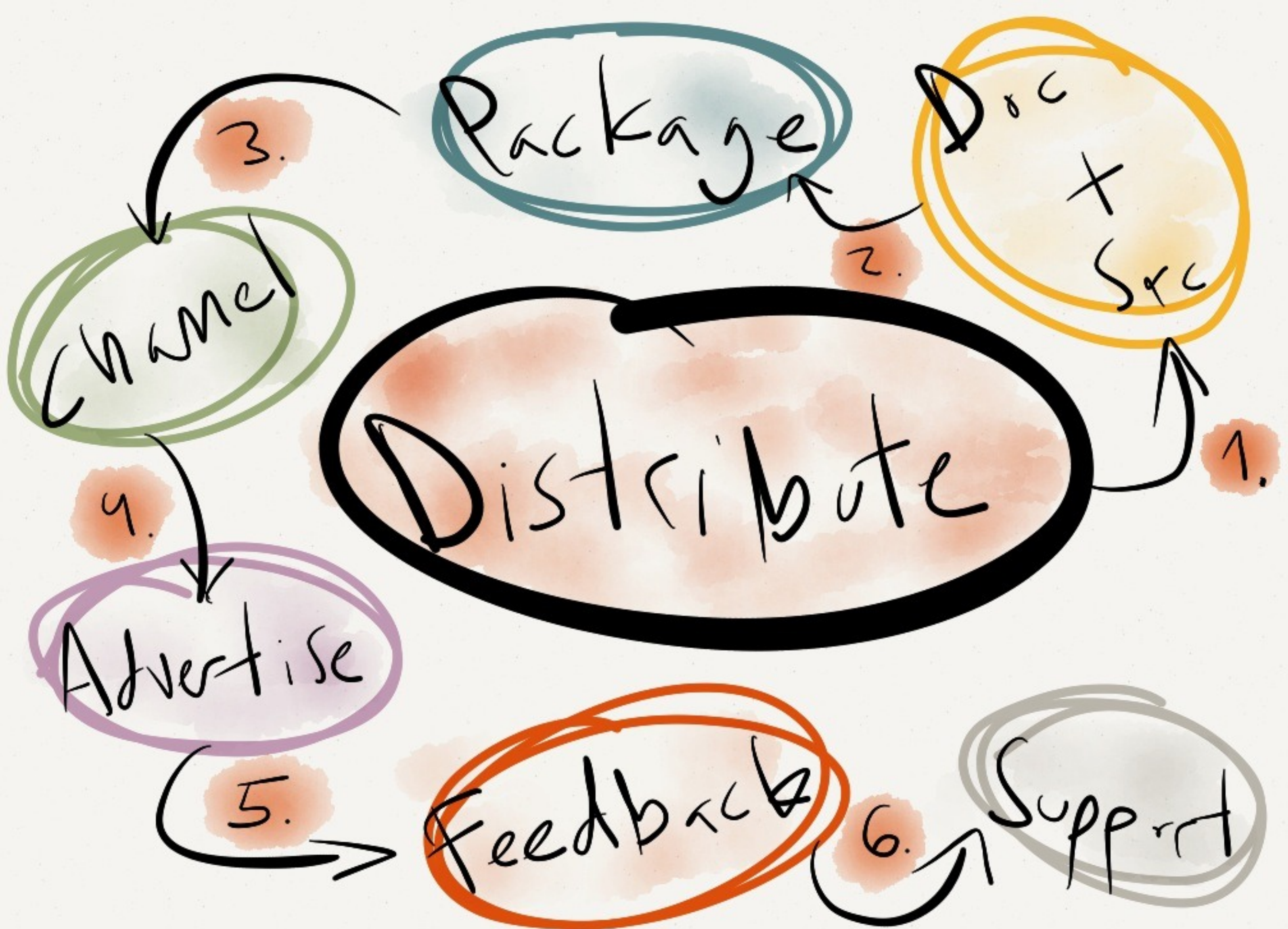
**There is no substitute
for manual testing and
user experience testing**

**Skip test driven
development, but invest
in regression testing**

**Pixel level regression
tests take time to set
up, but will be worth it**

**Never trust that
your changes would not
break other browsers
and skip cross-browser
testing**

It is impossible to use too much time in polishing UX for a reusable component.



😊 jojule / spreadsheet

🔧 Admin

👁 Unwatch

🍴 Fork

🏠 Pull Request

👁 1

🔧 1

Code

Network

Pull Requests 0

Issues 0

Wiki 0

Stats & Graphs

Simple spreadsheet component for Vaadin — [Read more](#)<https://vaadin.com/addon/spreadsheet>

🍏 Clone in Mac

📦 ZIP

SSH

HTTP

Git Read-Only

git@github.com:jojule/spreadsheet.git

📄

Read+Write access

🔼 branch: master ▾

Files

Commits

Branches 1

Tags

Downloads

🕒 Latest

Update



jojule

67b1

<https://github.com/jojule/spreadsheet>

spreadsheet /

| name | age | message | history |
|-------------------|--------------|-------------------------------------|---------|
| 📁 design | a day ago | Developing [Joonas Lehtinen] | |
| 📁 src | 19 hours ago | Version 0.1 [Joonas Lehtinen] | |
| 📄 README.markdown | 4 hours ago | Update README.markdown [jojule] | |
| 📄 licensing.txt | 19 hours ago | Version 0.1 [Joonas Lehtinen] | |
| 📄 pom.xml | 18 hours ago | Fixed add-on name [Joonas Lehtinen] | |

📖 README.markdown

Spreadsheet for Vaadin

The widget shows a spreadsheet - either from XLS file or by setting the cell contents programmatically.

This version is very limited and should be considered to be an early alpha -version. Try out the demo to see if it would be useful for you. I mainly built it for an upcoming presentation.

SpreadsheetView class should be also usable in GWT without Vaadin Framework, but then you must implement SpreadsheetModel by yourself.

Dependencies

- Apache POI 3.8 - <http://poi.apache.org/>
- Apache Commons Codec 1.5 - Required by POI - <http://commons.apache.org/codec/>

Release notes

Initial release with severe limitations:

- All columns and rows have fixed sizes
- No cell styling is supported
- No graphs are supported
- No merged cells are supported
- Performance for larger spreadsheets is really bad
- Only one spreadsheet widget is supported on screen at once

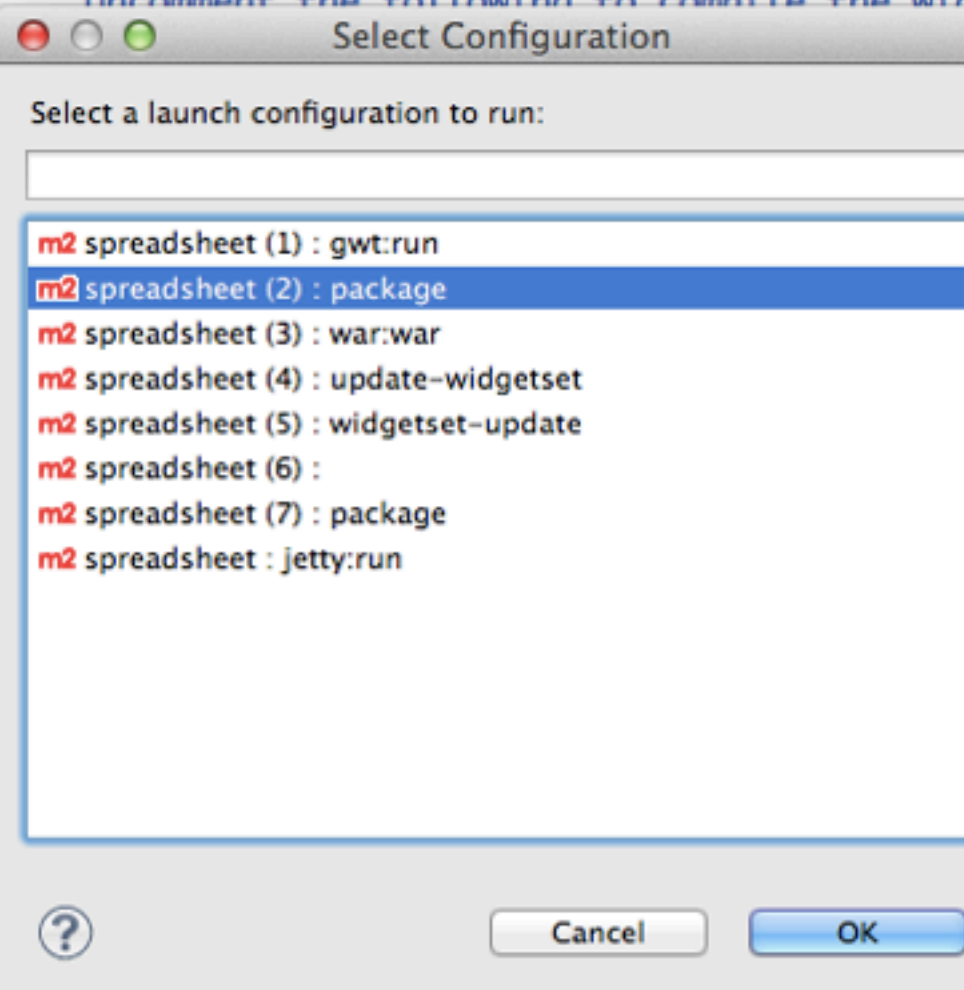
License & Author



SpreadsheetState.java SpreadsheetWidgetSet.gwt.xml

<!--

Uncomment the following to compile the widgetset for one browser only.
significantly when debugging.
Deployment to production



2.0.3 as a comma separated
list of writing were:

later and safari is used
Chrome.

</>

Design

Pro

spread

[INFO]

2012-04-15 20:14:54.406:INFO::jetty-6.1.24

JRebel: Directory '/Users/phoenix/dev/workspace/spreadsheet/target/classes' will be monitored for changes.

JRebel: Directory '/Users/phoenix/dev/workspace/spreadsheet/target/test-classes' will be monitored for changes.

2012-04-15 20:14:54.631:INFO::No Transaction manager found - if your webapp requires one, please configure one.

2012-04-15 20:14:54.997:INFO::Started SelectChannelConnector@0.0.0.0:8080

[INFO] Started Jetty Server

Apr 15, 2012 8:15:08 PM com.vaadin.terminal.gwt.server.AbstractApplicationServlet checkProductionMode

WARNING:

=====

Vaadin is running in DEBUG MODE.

Add productionMode=true to web.xml to disable debug features.

To show debug window, add ?debug to your application URL.


=====

versions/1.6.0/Home/bin/java (Apr 15, 2012 8:14:44 PM)

Vaadin Add-on Package Export

Define which resources should be exported into the Vaadin add-on package.

Select the resources to export.

☒ ▶  test

Manifest:

Implementation title:

Name of the add-on. Used in Vaadin Directory.

Implementation version:

Version of the addon. A "major.minor.revision" format is suggested.

Widgetsets:

Comma separated list of widgetsets included in the add-on. Refers to the GWT xml files (.gwt.xml).

Select the export destination:

JAR file:

Options:

☐ Overwrite existing files without warning

?

< Back

Next >

Cancel

Finish

Upload New Add-on

Select a category to post your new add-on to.

Note, that if you're updating a previous add-on, that is done by editing the add-on from the list above.



UI Components



Server-side and/or client-side
UI components



Themes



Themes for Vaadin applications



Data Components



Components related to the
Vaadin data model, e.g.
Container or Validator
implementations



Tools



Tools for Vaadin developers



Miscellaneous



Other Vaadin add-ons

Upload Add-on Package

Directory

All versions

Browse

All

[UI Components](#)
[Data Components](#)
[Themes](#)
[Tools](#)
[Miscellaneous](#)
[Official](#)
[Guest](#)
[Authoring](#)
[Subscribe RSS](#)
[Help](#)
[FAQ](#)
[Feedback](#)
[Most Recent](#) [Highest Rated](#) [Top Downloads](#)

Showing [CERTIFIED](#) [STABLE](#) [BETA](#) [EXPERIMENTAL](#)

« Previous [Next »](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [11](#) [12](#) ... [27](#)

316 Results

CSVValidation

In [UI Components](#) by [Marko Grönroos](#)

Client-side validation of text fields

Version 0.5.1 BETA

★★★★★ 3

↓ 1220

SimpleCalendar

In [UI Components](#) by [John Ahlroos](#)

Date selection made easy.

Version 0.1 EXPERIMENTAL No ratings yet

↓ 2

PopupButton

In [UI Components](#) by [Henri Kerola](#)

Button with popup

Version 2.2.0 BETA

★★★★★ 19

↓ 8546

Number Field

In [UI Components](#) by [Francesco Portus](#)

Numeric text field that allows insert only a valid numbers with keyboard and also allows spin value up/down.

Version 1.0.2 BETA

★★★★★ 2

↓ 99

ExternalLayout

In [UI Components](#) by [Risto Yrjänä](#)

ExternalLayout allows Vaadin components to be rendered outside the normal Vaadin application DOM-hierarchy

Version 2.0.1 STABLE

★★★★★ 1

↓ 186

FilterableTwinColSelect

In [UI Components](#) by [Kim Leppänen](#)

An improved version of the core's TwinColSelect, allows you to filter values

Version 1.0.0 EXPERIMENTAL No ratings yet

↓ 1

MockupContainer

In [Data Components](#) by [Risto Yrjänä](#)

Jain I18N

In [Data Components](#) by [Lokesh Jain](#)

Directory

Browse

All

[UI Components](#)[Data Components](#)[Themes](#)[Tools](#)[Miscellaneous](#)[Guest](#)[Authoring](#)[Subscribe RSS](#)[Help](#)[FAQ](#)[Feedback](#)

PaperStack

In [UI Components](#) by [Tomi Virkki](#) ★★★★★ 11 ↓ 194

[Report this add-on](#)**Version**

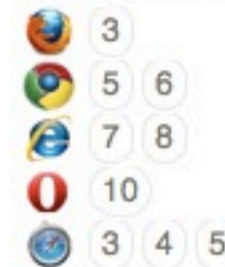
0.8.1 (latest)

Maturity

EXPERIMENTAL

License[Apache License 2.0](#)**Vaadin**

6.2 upwards

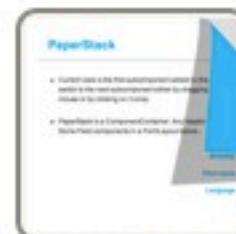
Browser Compatibility

Overview

PaperStack is a component container whose subcomponents are presented sequentially, one subcomponent at a time. User can switch between the subcomponents by mouse dragging the upper right corner of a view revealing the underlying subcomponent simultaneously. The transition effect simulates leafing through a stack of papers.

Highlights

```
1 package org.vaa
2
3 import com.vaad
4 import com.vaad
5
6 public class MyA
```

[Code Example](#)[Screenshot 2](#)[Screenshot 1](#)

Release notes

0.8.1:

Download Now

Version 0.8.1 (86 kB)

**Maven POM**

Related Links

- [Discussion Forum](#)
- [Online Demo](#)
- [Source Code](#)

Share

 | [More...](#)[Permalink to this add-on:](#)<http://vaadin.com/addon/paperstack>

Directory

Browse

All

UI Components

Data Components

Themes

Tools

Miscellaneous

Guest

[Authoring](#)

[Subscribe RSS](#)

[Help](#)

[FAQ](#)

[Feedback](#)

PaperStack

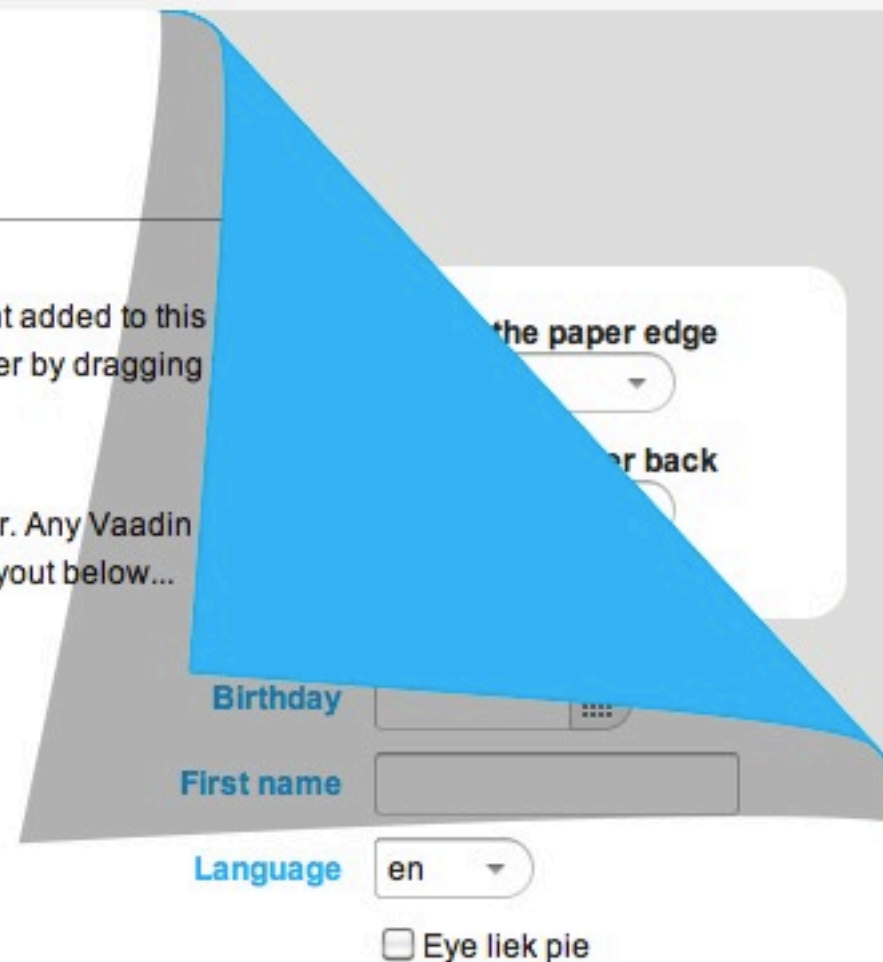
by [Tom Middeldijk](#) 44 5 404

[Report this add-on](#)

PaperStack - Screenshot 1

PaperStack

- Current view is the first subcomponent added to this switch to the next subcomponent either by dragging mouse or by clicking on it once.
- PaperStack is a ComponentContainer. Any Vaadin Some Field components in a FormLayout below...



5
6 `public class MyA`

[Code Example](#)



[Screenshot 2](#)



[Screenshot 1](#)

Release notes

0.0.1:



[+ More...](#)

on:

[n/paperstack](#)

Directory

[Browse](#)[All](#)[UI Components](#)[Data Components](#)[Themes](#)[Tools](#)[Miscellaneous](#)[Guest](#)[Authoring](#)[Subscribe RSS](#)[Help](#)[FAQ](#)[Feedback](#)

PaperStack

In [UI Components](#) by [Tomi Virkki](#) ★★★★★ 11 ↓ 194

[Report this add-on](#)

Version

0.8.1 (latest)

Maturity

EXPERIMENTAL

Browser Compatibility



Download Now

Version 0.8.1 (86 kB)



Maven POM



PaperStack - Code Example

```
PaperStack paperStack = new PaperStack();

paperStack.addComponent(new Label("Hello!"));

InlineDateField inlineDateField = new InlineDateField();
inlineDateField.setResolution(DateField.RESOLUTION_DAY);
paperStack.addComponent(inlineDateField, "#999");

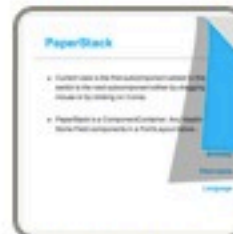
mainWindow.addComponent(paperStack);
```

```
1 package org.vaa
2
3 import com.vaad
4 import com.vaad
5
6 public class MyA
```

Code Example



Screenshot 2



Screenshot 1

Release notes

0.8.1:

Related Links

[Discussion Forum](#)
[Online Demo](#)
[Source Code](#)

e

[More...](#)

Permalink to this add-on:

<http://vaadin.com/addon/paperstack>

Directory

Browse

All

[UI Components](#)[Data Components](#)[Themes](#)[Tools](#)[Miscellaneous](#)[Guest](#)[Authoring](#)[Subscribe RSS](#)[Help](#)[FAQ](#)[Feedback](#)

PaperStack

In [UI Components](#) by [Tomi Virkki](#) ★★★★★ 11 ↓ 194[Report this add-on](#)

Version

0.8.1 (latest)

Maturity

EXPERIMENTAL

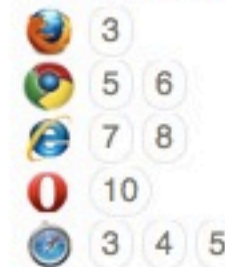
License

[Apache License 2.0](#)

Vaadin

6.2 upwards

Browser Compatibility



Overview

PaperStack is a component container whose subcomponents are presented sequentially, one subcomponent at a time. User can switch between the subcomponents by mouse dragging the upper right corner of a view revealing the underlying subcomponent simultaneously. The transition effect simulates leafing through a stack of papers.

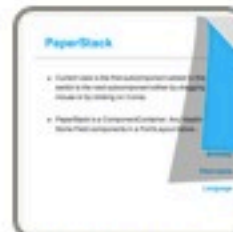
Highlights

```
1 package org.vaa
2
3 import com.vaad
4 import com.vaad
5
6 public class MyA
```

Code Example



Screenshot 2



Screenshot 1

Release notes

0.8.1:

[Download Now](#)

Version 0.8.1 (86 kB)

[Maven POM](#)

```
<dependency>
  <groupId>org.vaadin.addons</groupId>
  <artifactId>paperstack</artifactId>
  <version>0.8.1</version>
</dependency>
```

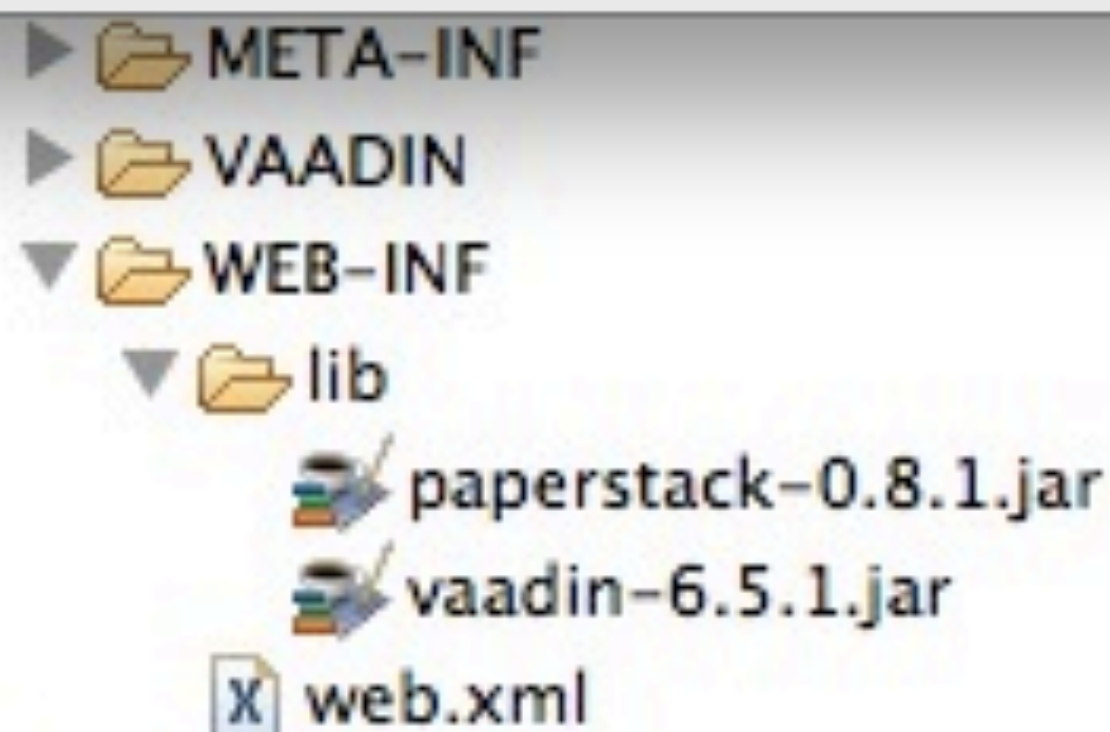
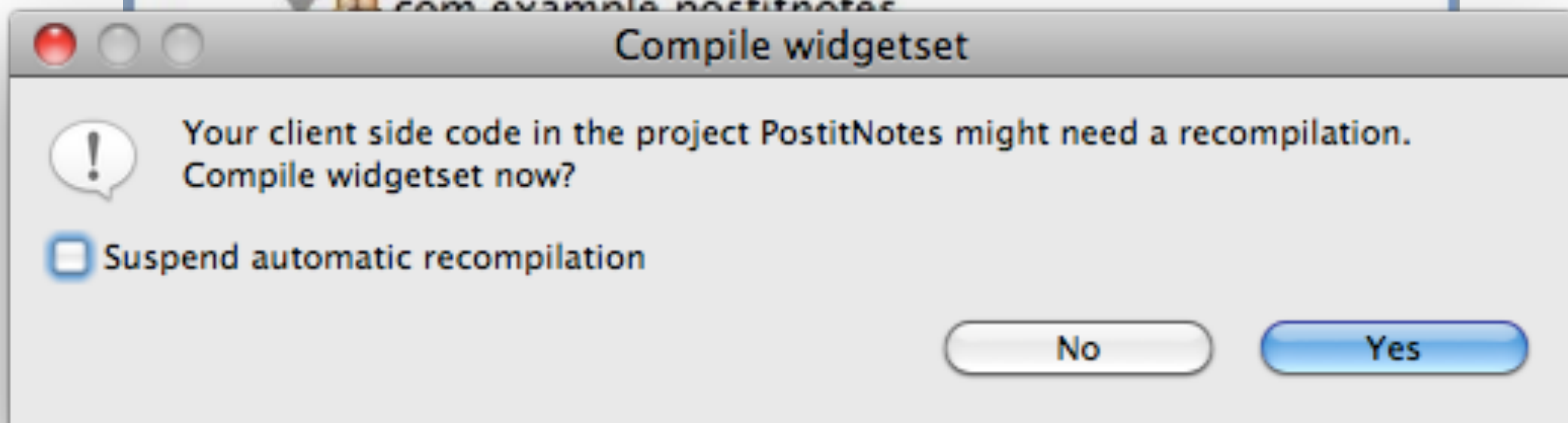
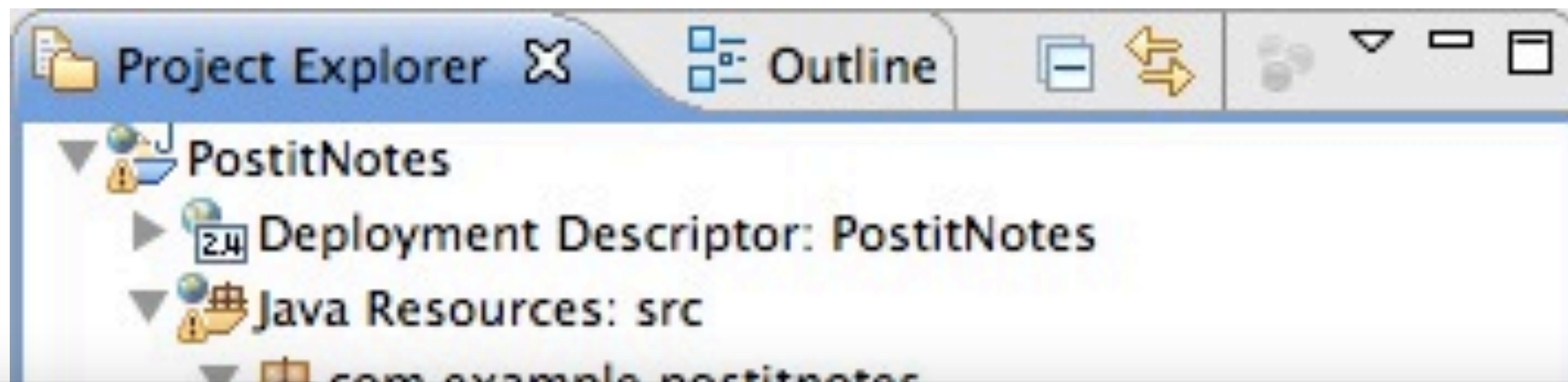
```
<repository>
  <id>vaadin-addons</id>
  <url>http://maven.vaadin.com/vaadin-
  addons</url>
</repository>
```

Related Links

- [Discussion Forum](#)
- [Online Demo](#)
- [Source Code](#)

Share

[Permalink to this add-on:](#)



Try out my Spreadsheet!

vaadin.com/addon/spreadsheet



ED00864B

PATTISON

Issues · jojule/spreadsheet

https://github.com/jojule/spreadsheet/issues GitHub, Inc. RSS Google

github Search... Explore Gist Blog Help jojule 35

😊 jojule / spreadsheet Admin Unwatch Fork Pull Request 1 1

Code Network Pull Requests 0 Issues 0 Wiki 0 Stats & Graphs

Browse Issues Milestones Search: Issues & Milestones... New Issue

Everyone's Issues Assigned to you Mentioning you

No milestone selected

Labels

| | |
|-------------|---|
| bug | 0 |
| duplicate | 0 |
| enhancement | 0 |
| invalid | 0 |
| question | 0 |
| wontfix | 0 |

Manage Labels

keyboard shortcuts available

Updated Comments

<https://github.com/jojule/spreadsheet/issues>

New Issue – jojule/spreadsheet

◀▶

+

🔍

https://github.com/jojule/spreadsheet/issues/new

GitHub, Inc.

RSS

↺

🔍

Google

⬇

Code

Network

Pull Requests0

Issues0

Wiki0

Stats & Graphs

Browse Issues

Milestones

Search: Issues & Milestones...🔍

New Issue

Title

Does not work properly if there are two instances of the component

No one is assigned⚙️

No milestone⚙️

Write

Preview

Comments are parsed with GitHub Flavored Markdown

When two instances of the Spreadsheet are put to screen at the same time, scrolling either scrolls the headers in both. This makes the component unusable for any screens with two spreadsheets.

Add Labels

bug

duplicate

enhancement

invalid

question

wontfix

Submit new Issue

Issues · jojule/spreadsheet

https://github.com/jojule/spreadsheet/issues

GitHub, Inc. RSS

Google

github

Search...

Explore Gist Blog Help

jojule

35

😊 jojule / spreadsheet

Admin

Unwatch

Fork

Pull Request

1

1

Code

Network

Pull Requests 0

Issues 1

Wiki 0

Stats & Graphs

Browse Issues

Milestones

Search: Issues & Milestones...

New Issue

Everyone's Issues 1

Assigned to you 0

Mentioning you 0

No milestone selected

Labels

bug 1

duplicate 0

enhancement 0

invalid 0

question 0

wontfix 0

Manage Labels

No active filters. Use the sidebar to filter issues.

Keyboard shortcuts available

1 open issue

0 closed issues

Submitted

Updated

Comments

Close

Label ▾

Assignee ▾

Milestone ▾

#1

Does not work properly if there are two instances of the component


bug

by jojule just now

1 open issue in this view

Support HOWTO

**Issue
reported by
actual user!**



```
graph LR; A[Issue reported by actual user!] --> B[Ignore. wont-fix in best case. (the usual open source way)]; A --> C[Fix after 6 months, maybe...]; A --> D[Fix immediately and thank the user who reported it];
```

Ignore. wont-fix in best case.
(the usual open source way)

Fix after 6 months,
maybe...

**Fix immediately and thank
the user who reported it**



Tue **Vaadin &**
20:00 C4 **GWT BOF**

Wed **Intro to**
14:00 C4 **Vaadin 7**

joonas@vaadin.com
vaadin.com/joonas
@joonaslehtinen