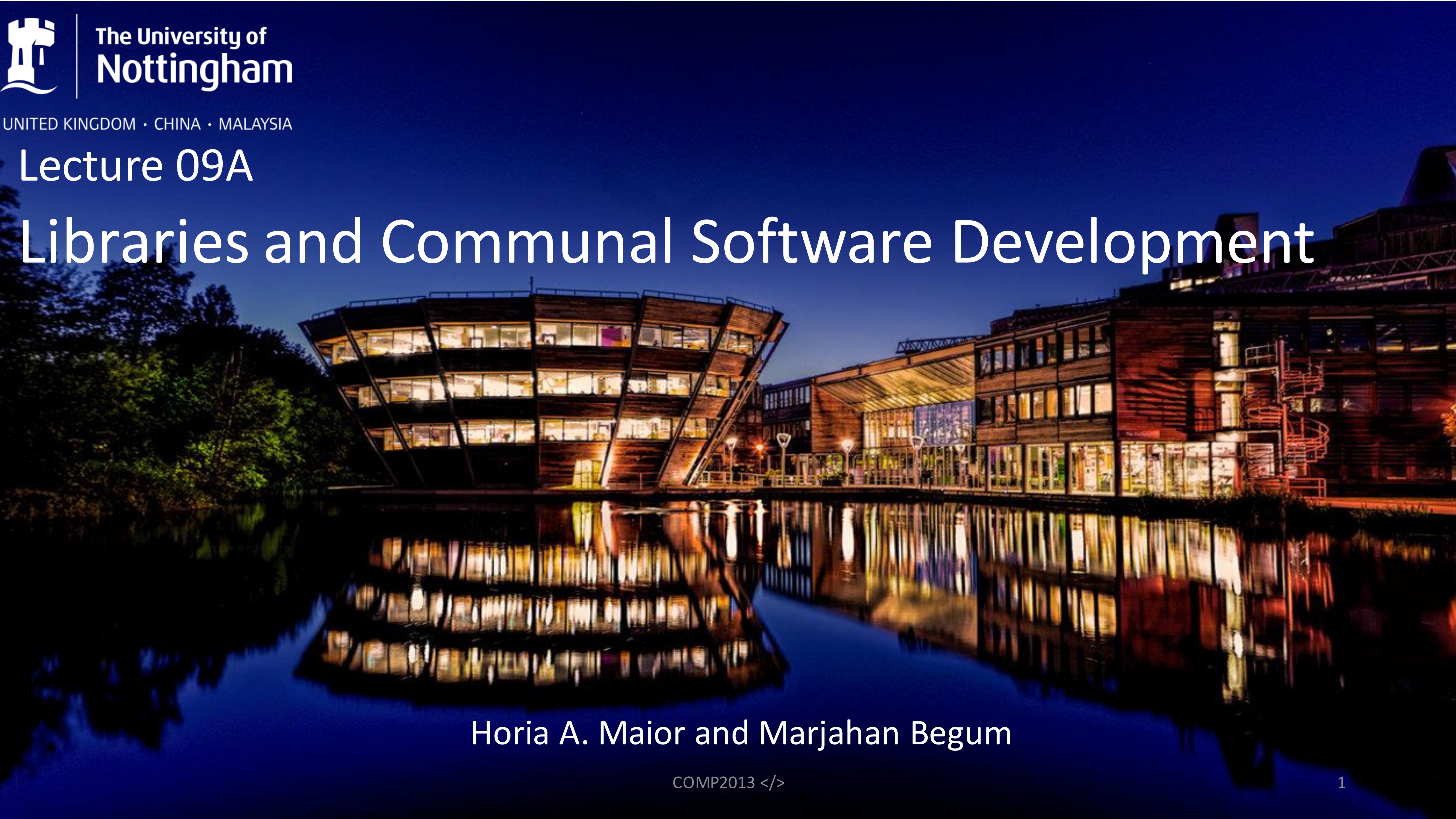


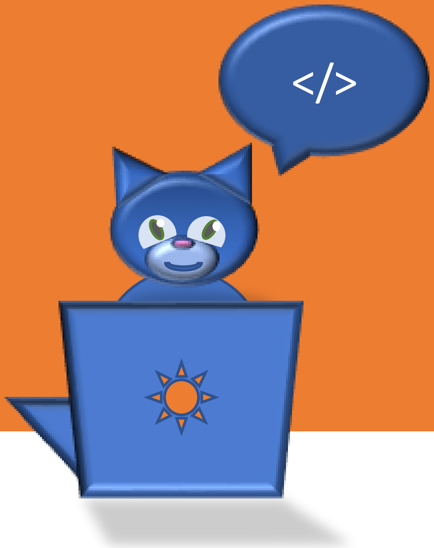
The University of Nottingham

UNITED KINGDOM · CHINA · MALAYSIA

Lecture 09A

Libraries and Communal Software Development

Horia A. Maior and Marjahan Begum



COMP2013
Software Maintenance

Lecture 09A

Libraries and Communal Software Development

Horia A. Maior and Marjahan Begum

University of Nottingham
UK | CHINA | MALAYSIA

Topics

</>

- Lecture
  - Coursework tips
  - Using and creating libraries
  - Communal software development (what it is; licensing; how to do it; how to get involved)
- Lab
  - Time for you to work on your coursework
- Lecture
  - Time for you to work on your coursework

</>

coursework tips

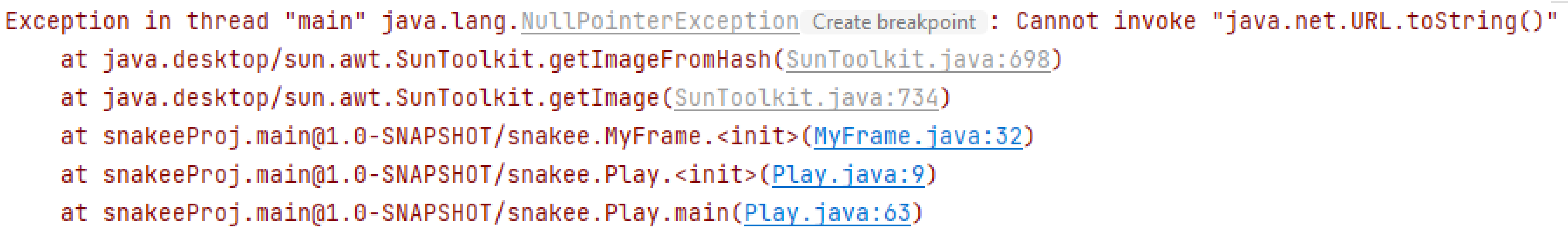
Screen Capture Demos on Moodle

There are many useful screen capture demonstration videos in the lectures and on Moodle:

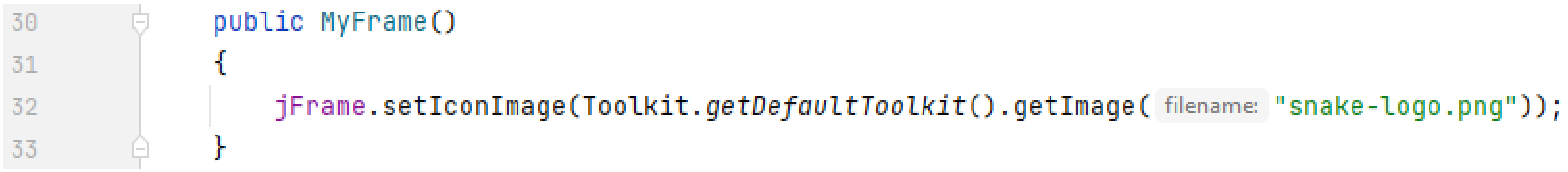
- Setting Up IntelliJ
- Setting Up Git and GitLab
- Maven/Gradle Project
- JavaFX Project
- Setting Up Junit testing within a JavaFX project using Maven (later today)
- Early set up of the CW

Path Issues

- An error message like this indicates that the compiler cannot find an image
    - Usually the path is set wrongly

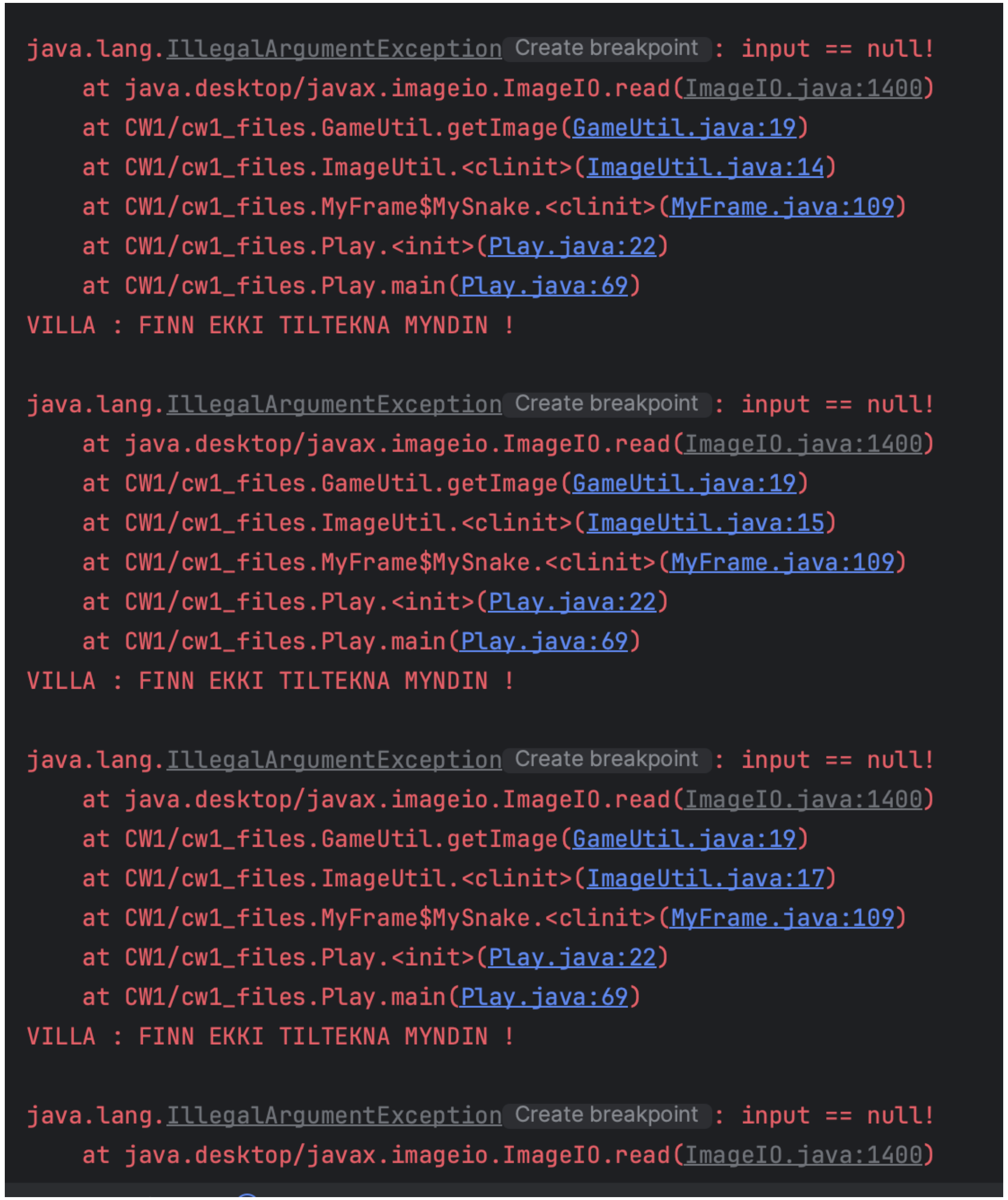


```
Exception in thread "main" java.lang.NullPointerException Create breakpoint : Cannot invoke "java.net.URL.toString()"
    at java.desktop/sun.awt.SunToolkit.getImageFromHash(SunToolkit.java:698)
    at java.desktop/sun.awt.SunToolkit.getImage(SunToolkit.java:734)
    at snakeeProj.main@1.0-SNAPSHOT/snakee.MyFrame.<init>(MyFrame.java:32)
    at snakeeProj.main@1.0-SNAPSHOT/snakee.Play.<init>(Play.java:9)
    at snakeeProj.main@1.0-SNAPSHOT/snakee.Play.main(Play.java:63)
```



```
30    public MyFrame()
31    {
32        jFrame.setIconImage(Toolkit.getDefaultToolkit().getImage( filename: "snake-logo.png"));
33    }
```

- Likely to do with the same issue the paths to resources folder.

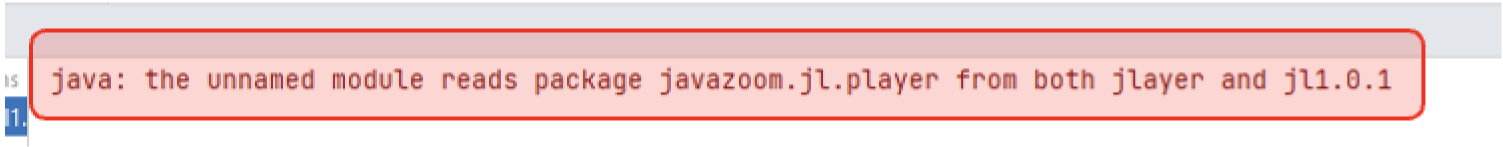


```
java.lang.IllegalArgumentException Create breakpoint : input == null!
    at java.desktop/javax.imageio.ImageIO.read(ImageIO.java:1400)
    at CW1/cw1_files.GameUtil.getImage(GameUtil.java:19)
    at CW1/cw1_files.ImageUtil.<clinit>(ImageUtil.java:14)
    at CW1/cw1_files.MyFrame$MySnake.<clinit>(MyFrame.java:109)
    at CW1/cw1_files.Play.<init>(Play.java:22)
    at CW1/cw1_files.Play.main(Play.java:69)
VILLA : FINN EKKI TILTEKNA MYNDIN !

java.lang.IllegalArgumentException Create breakpoint : input == null!
    at java.desktop/javax.imageio.ImageIO.read(ImageIO.java:1400)
    at CW1/cw1_files.GameUtil.getImage(GameUtil.java:19)
    at CW1/cw1_files.ImageUtil.<clinit>(ImageUtil.java:15)
    at CW1/cw1_files.MyFrame$MySnake.<clinit>(MyFrame.java:109)
    at CW1/cw1_files.Play.<init>(Play.java:22)
    at CW1/cw1_files.Play.main(Play.java:69)
VILLA : FINN EKKI TILTEKNA MYNDIN !

java.lang.IllegalArgumentException Create breakpoint : input == null!
    at java.desktop/javax.imageio.ImageIO.read(ImageIO.java:1400)
    at CW1/cw1_files.GameUtil.getImage(GameUtil.java:19)
    at CW1/cw1_files.ImageUtil.<clinit>(ImageUtil.java:17)
    at CW1/cw1_files.MyFrame$MySnake.<clinit>(MyFrame.java:109)
    at CW1/cw1_files.Play.<init>(Play.java:22)
    at CW1/cw1_files.Play.main(Play.java:69)
VILLA : FINN EKKI TILTEKNA MYNDIN !

java.lang.IllegalArgumentException Create breakpoint : input == null!
    at java.desktop/javax.imageio.ImageIO.read(ImageIO.java:1400)
```

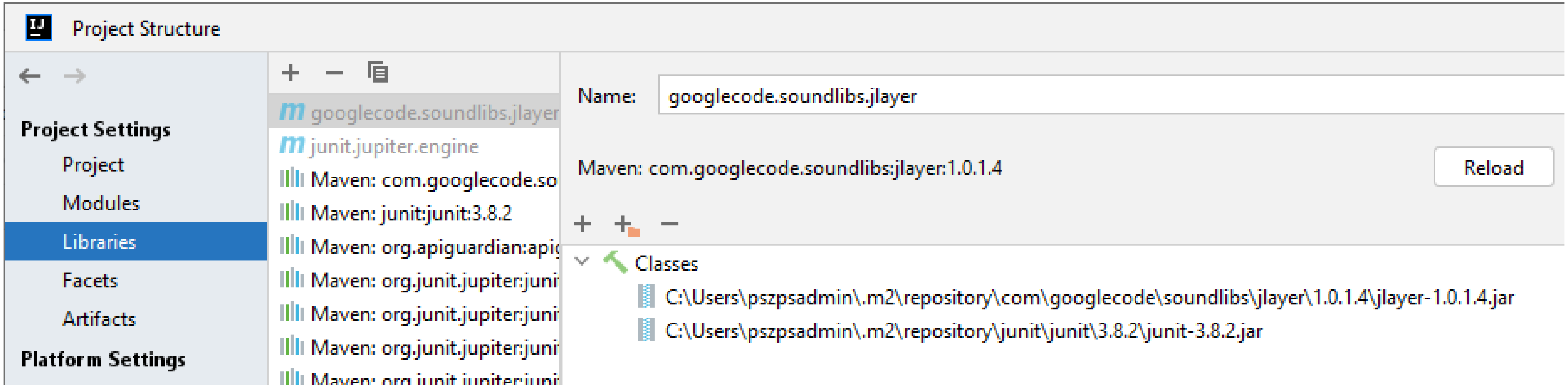
Maven. How to use a POM file?

- Project Object Model (POM)
- Contains configuration information about your project
- Key items
  - Project Identification
  - Dependencies
  - Plug-Ins
  - Other Settings

Error: Reading packages from multiple sources



```
java: the unnamed module reads package javazoom.jl.player from both jlayer and jl1.0.1
```



googlecode.soundlibs.jlayer
junit.jupiter.engine
Maven: com.googlecode.so
Maven: junit:junit:3.8.2
Maven: org.apiguardian:apig
Maven: org.junit.jupiter:juni
Maven: org.junit.jupiter:juni
Maven: org.junit.jupiter:juni
Maven: org.junit.jupiter:juni

Name: googlecode.soundlibs.jlayer

Maven: com.googlecode.soundlibs:jlayer:1.0.1.4

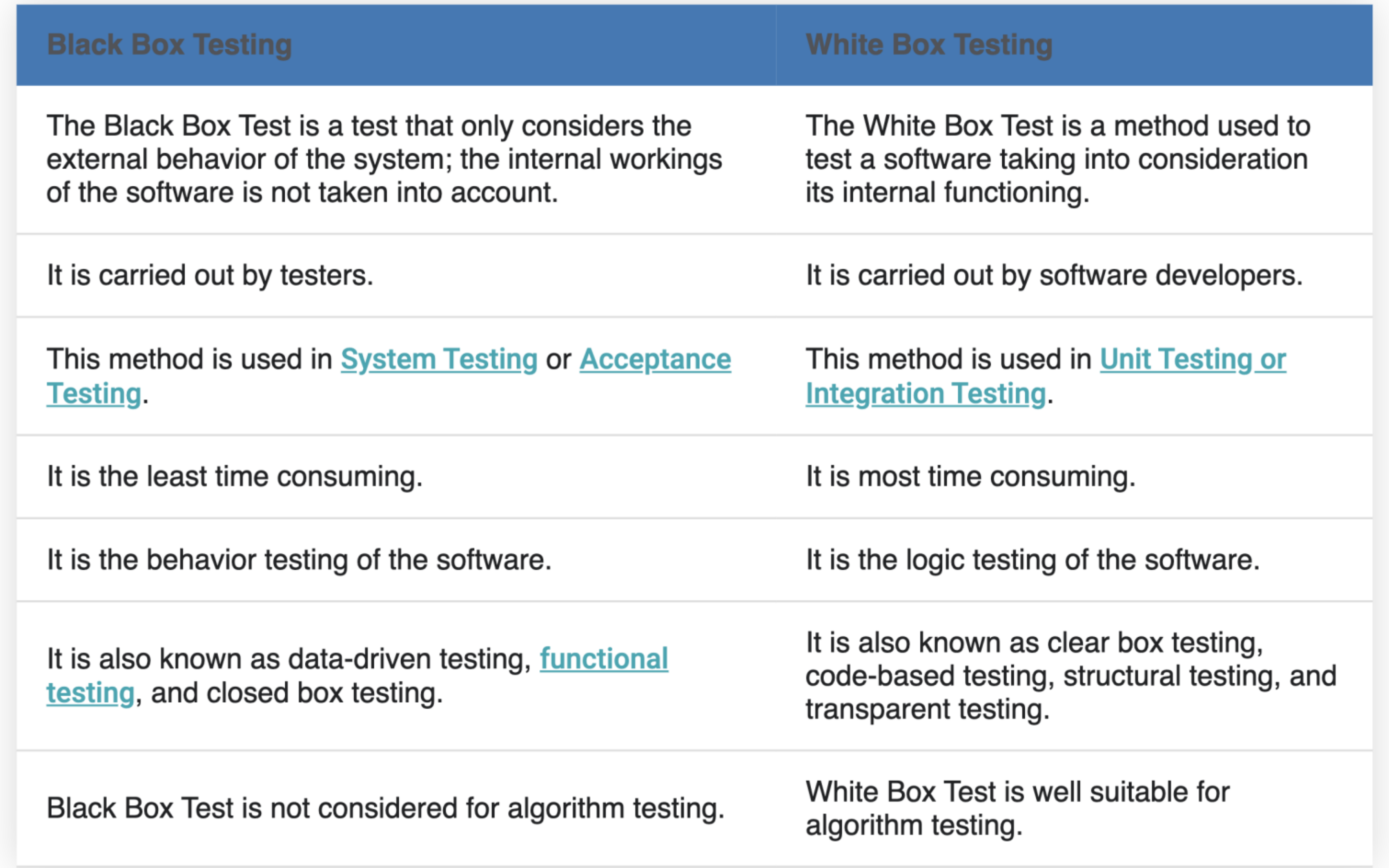
C:\Users\pszpsadmin\.m2\repository\com\googlecode\soundlibs\jlayer\1.0.1.4\jlayer-1.0.1.4.jar
C:\Users\pszpsadmin\.m2\repository\junit\junit\3.8.2\junit-3.8.2.jar

Random Tips

- Consider the use of "streams" and "lambdas" for improving maintainability
  - https://stackify.com/streams-guide-java-8/
- Solving problems is detective work
  - Check out "Stack Exchange"
  - Use prototyping and throw away models
  - Use the debugger to get information about the state of variables
    - https://youtube.com/playlist?list=PLPZy-hmwOdEUWF85MuwrKV8YVWLmZW4ZA&si=gj8AmbZVC9q_xQMY
  - Look at the examples from the lectures and on Moodle
- When submitting your coursework
  - Don't forget to merge back to the main branch
  - Don't forget to tidy up your project by using "mvn clean" or "gradle clean" before zipping it

Random Tips

- Feel free to use your own graphics/sound
- Testing plan/strategy
  - White Box
  - Black Box
  - Keep Track of Testing



Black Box Testing	White Box Testing
The Black Box Test is a test that only considers the external behavior of the system; the internal workings of the software is not taken into account.	The White Box Test is a method used to test a software taking into consideration its internal functioning.
It is carried out by testers.	It is carried out by software developers.
This method is used in System Testing or Acceptance Testing.	This method is used in Unit Testing or Integration Testing.
It is the least time consuming.	It is most time consuming.
It is the behavior testing of the software.	It is the logic testing of the software.
It is also known as data-driven testing, functional testing, and closed box testing.	It is also known as clear box testing, code-based testing, structural testing, and transparent testing.
Black Box Test is not considered for algorithm testing.	White Box Test is well suitable for algorithm testing.

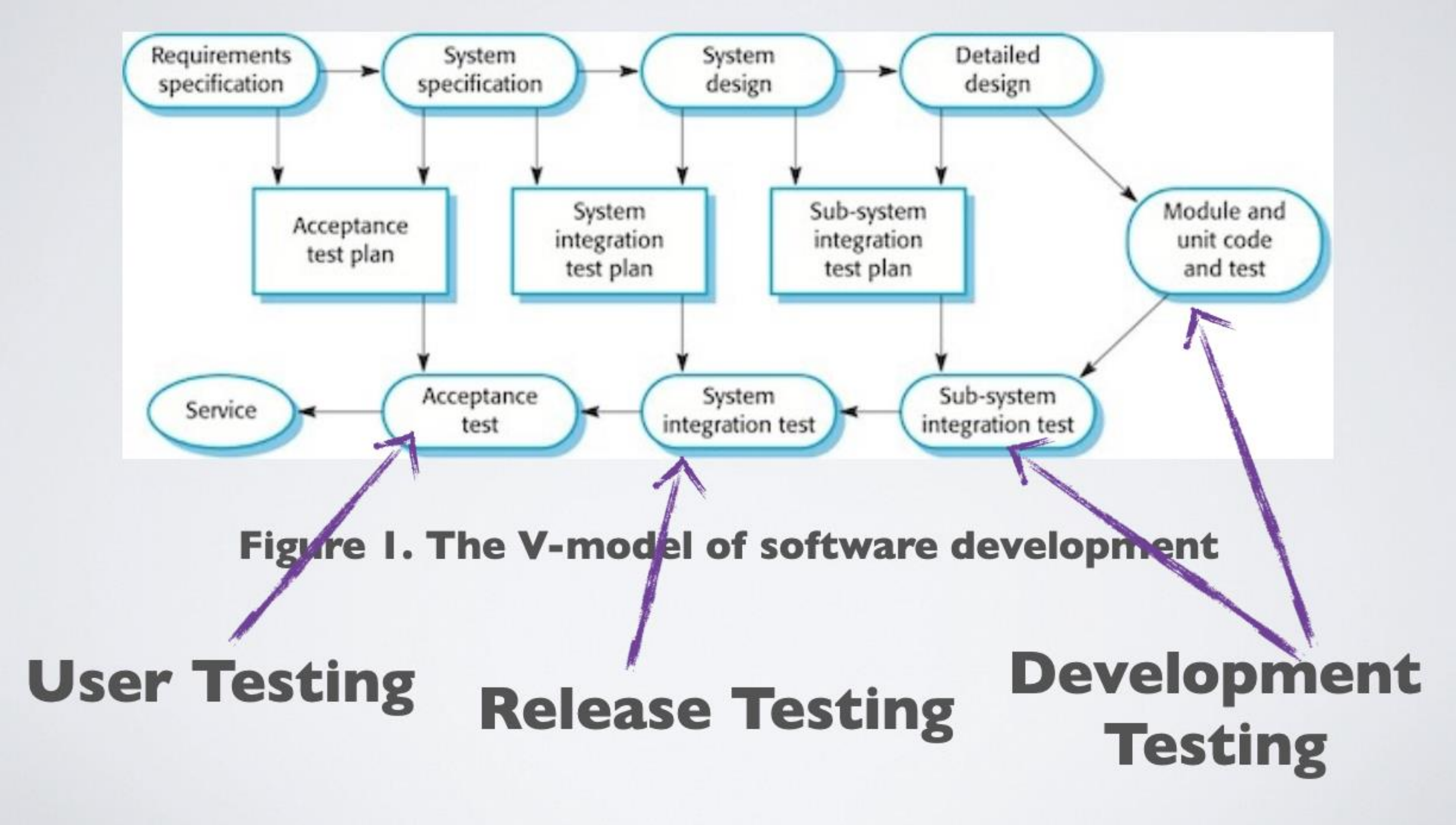
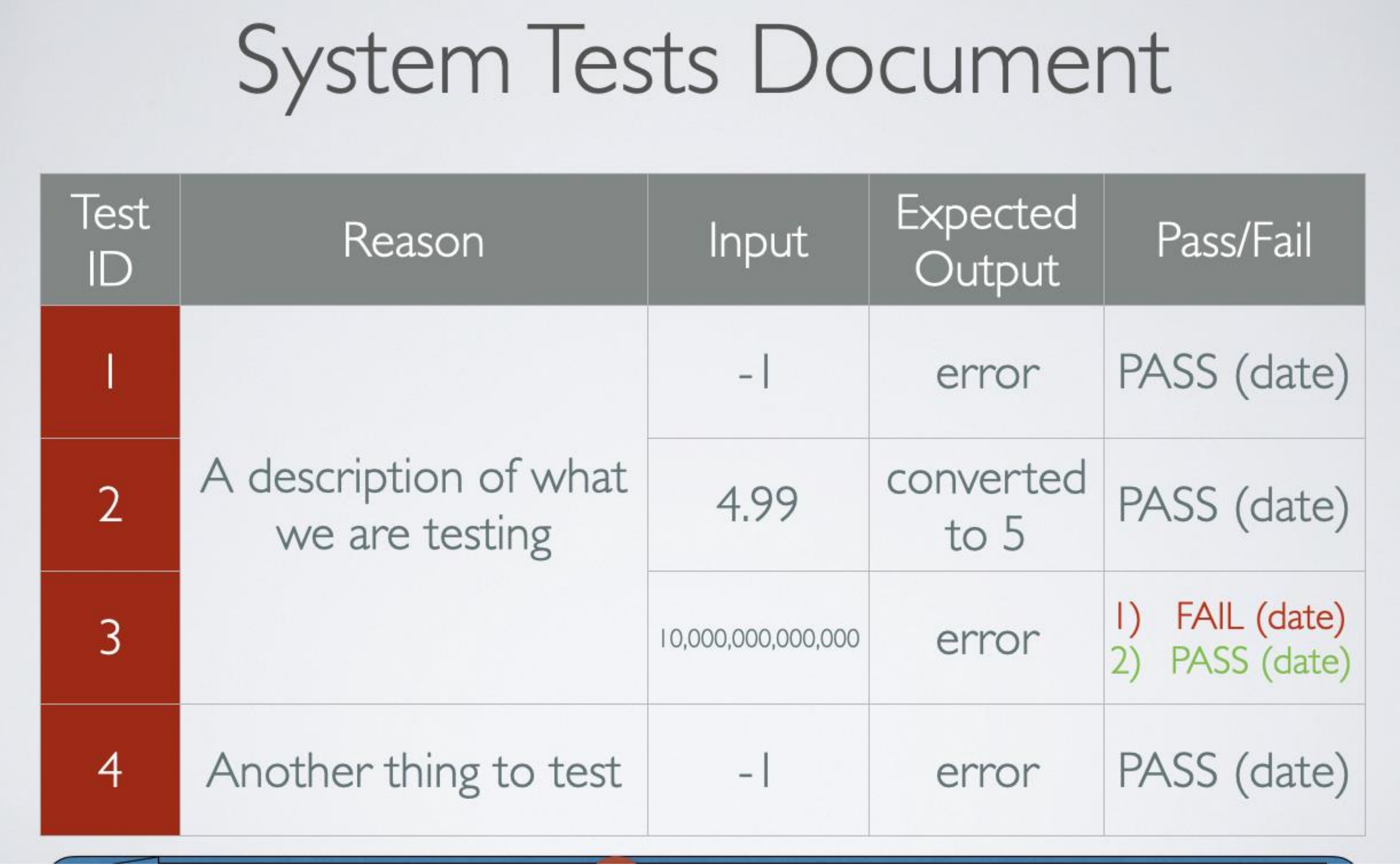

Requirements specification
System specification
System design
Detailed design
Acceptance test plan
System integration test plan
Sub-system integration test plan
Module and unit code and test
Service
Acceptance test
System integration test
Sub-system integration test

Figure 1. The V-model of software development

User Testing

Release Testing

Development Testing



System Tests Document

Test ID	Reason	Input	Expected Output	Pass/Fail
1	A description of what we are testing	-1	error	PASS (date)
2		4.99	converted to 5	PASS (date)
3		10,000,000,000,000	error	1) FAIL (date) 2) PASS (date)
4	Another thing to test	-1	error	PASS (date)

Useful Resources

- Maven tutorial Crash Course https://www.youtube.com/watch?v=Xatr8AZLOsE
- Gradle tutorial Crash Course https://www.youtube.com/watch?v=gKPMKRnnbXU
- JUnit 5 tutorial Crash Course https://www.youtube.com/watch?v=6uSnF6IuWIw

philosophies of software development

Different Philosophies of Software Development

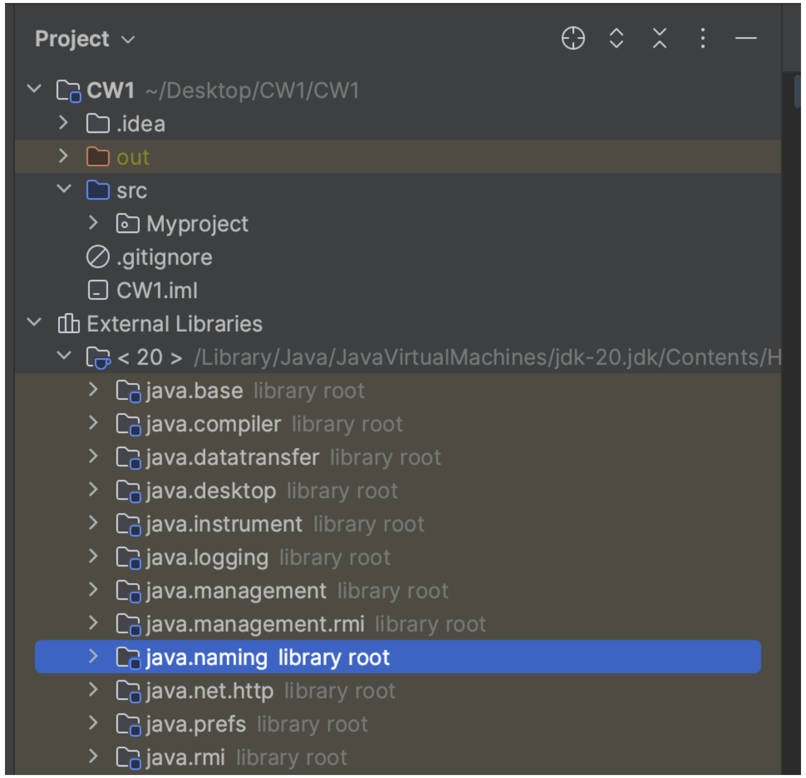
</>

- During your life as a software maintainer you will work on a number of different forms of projects
- Some of these may have open source code, or use open source libraries
- You may have to rework private code to be open source, or vice versa
- In this lecture, we will look at an overview of how to use third party code in the form of libraries and open source projects

libraries

What is a Library?

- What is a library?
  - Some 3rd party software packaged up (in binaries) and ready-to-use in your own code
  - It is a shared resource
- Usually online documentation
  - Supporting guides as well as Javadocs to show the API
- You've already had experience of this
  - e.g. using the JDK library in your IDE



```
Project
CW1  ~/Desktop/CW1/CW1
  .idea
  out
  src
    Myproject
  .gitignore
  CW1.iml
External Libraries
  < 20 >  /Library/Java/JavaVirtualMachines/jdk-20.jdk/Contents/H
    java.base  library root
    java.compiler  library root
    java.datatransfer  library root
    java.desktop  library root
    java.instrument  library root
    java.logging  library root
    java.management  library root
    java.management.rmi  library root
    java.naming  library root
    java.net.http  library root
    java.prefs  library root
    java.rmi  library root
```

What's in a Library?

- Libraries in Java consists of:
  - A Jar file
    - Created via the jar tool, or via an IDE
    - Basically a zip file
    - A way of packaging class files and resource files
    - Contains a special folder – META-INF
    - Note that a runnable Jar file is not a library!
  - An API
    - Publicly accessible methods
    - Interface stability is important; use "@Deprecated" before removing methods
      - In Javadoc use "@deprecated"
  - Usually include a licence
    - How you can distribute/change it – more later

Making Use of Libraries in Java

- You need a library file - usually a .jar archive
- Reference it in your project
  - e.g. adding the jar to your Java Build Path
- Import relevant parts of the library into your code
- Make use of the methods
  - May need to create an object, or static use access
- Think about how to distribute the library with your code

Making Use of Libraries in Java

</>

- You can package them up with your deployed application, or perhaps include them in your project source distribution
  - Licence permitting! (more later…)

- You can also use build files to help with collaborative development using libraries
  - Script will copy a file from an external resource
  - In this way you do not need a license to re-distribute them as you are only downloading them

Build Systems and Libraries

- Build systems can pull required libraries from remote sources
- Maven/Gradle will place these dependences in a folder locally:

  {your-username}\.m2

  {your-username}\.gradle

- You can request particular versions of libraries, so you may see different versions of libraries in this folder

open source software development and maintenance

Open Source Software

- What is Open Source Software (OSS)?
  - OSS is (generally) free software that uses any licence approved by the Open Source Initiative (OSI) from their list of approved open source licences (link below)

- What is Free OSS?
  - "Software that gives users rights to run, copy, distribute, change and improve it as they see it, without them asking permission from or make payments to any external group or person".

Mitre FOSS report 2002

- Open source initiative: https://opensource.org/

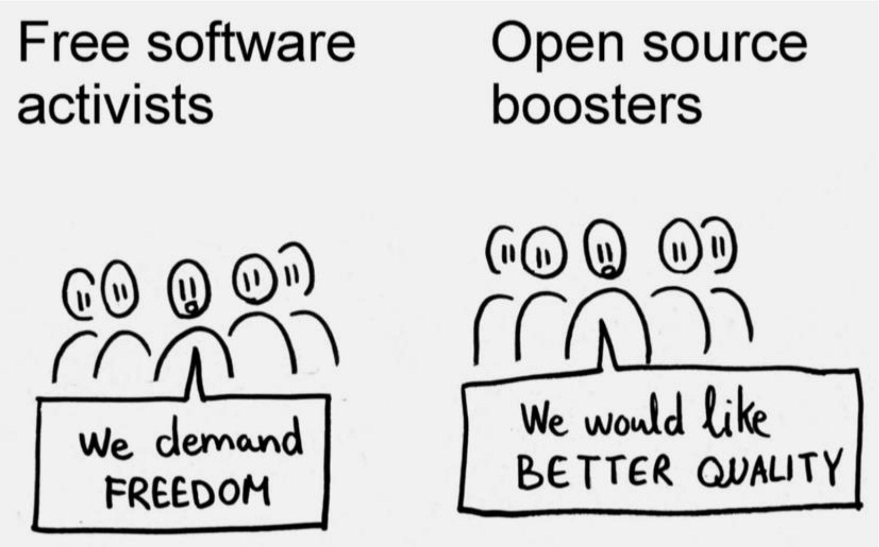
open source initiative®

Free Software Foundation

- Richard Stallman: President and founder of the Free Software Foundation

"Free software should be a human right!"

Different perspectives



Free software, free society: Richard Stallman at TEDxGeneva 2014
https://www.youtube.com/watch?v=Ag1AKIl_2GM

Open Source Software

- Why Go Open Source?
  - Higher Quality
  - Customisable
  - Improvable
  - Collaborative bug finding/fixing
  - Redistributable
  - Transparency
  - Free

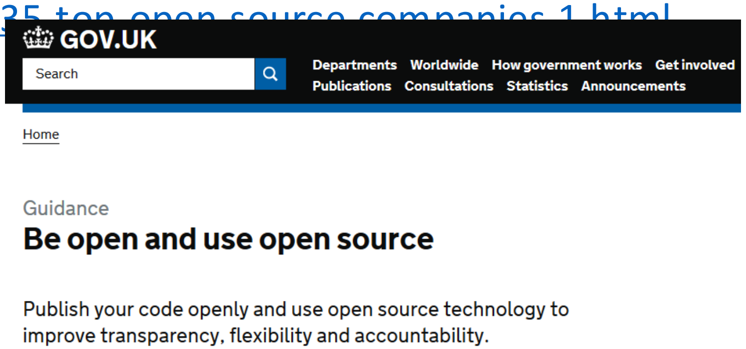
Open Source Software

- Starting to be recognised as a "Good Thing" by industry. Why?
  - All the previous reasons plus:

1. Good advertising
2. Attract talented developers
   a) More development possible
   b) Version-tracked contributions are good demonstration of potential employees’ programming skill
   c) Develop outside of your core skill set

Open Source Software

- Examples:
  - Linux, OpenJDK; Apache; LaTex; Moodle; Firefox; Android; Mozilla; MySQL; OpenOffice; Blender; VLC; IntelliJ CE; Eclipse
- Top "Open Source" Companies
  - Adobe; Amazon Web Services; Docker; Facebook; GitHub; Google; Gradle; Huawei; IBM; Intel; LinkedIn; Microsoft; MongoDB; Netflix; Oracle; Red Hat; Samsung Electronics; Twitter; ...
    - More information about how these companies contribute is available here:
      - https://www.datamation.com/open-source/35-top-open-source-companies-1.html
- The UK government supports OSS,
  - They should do!



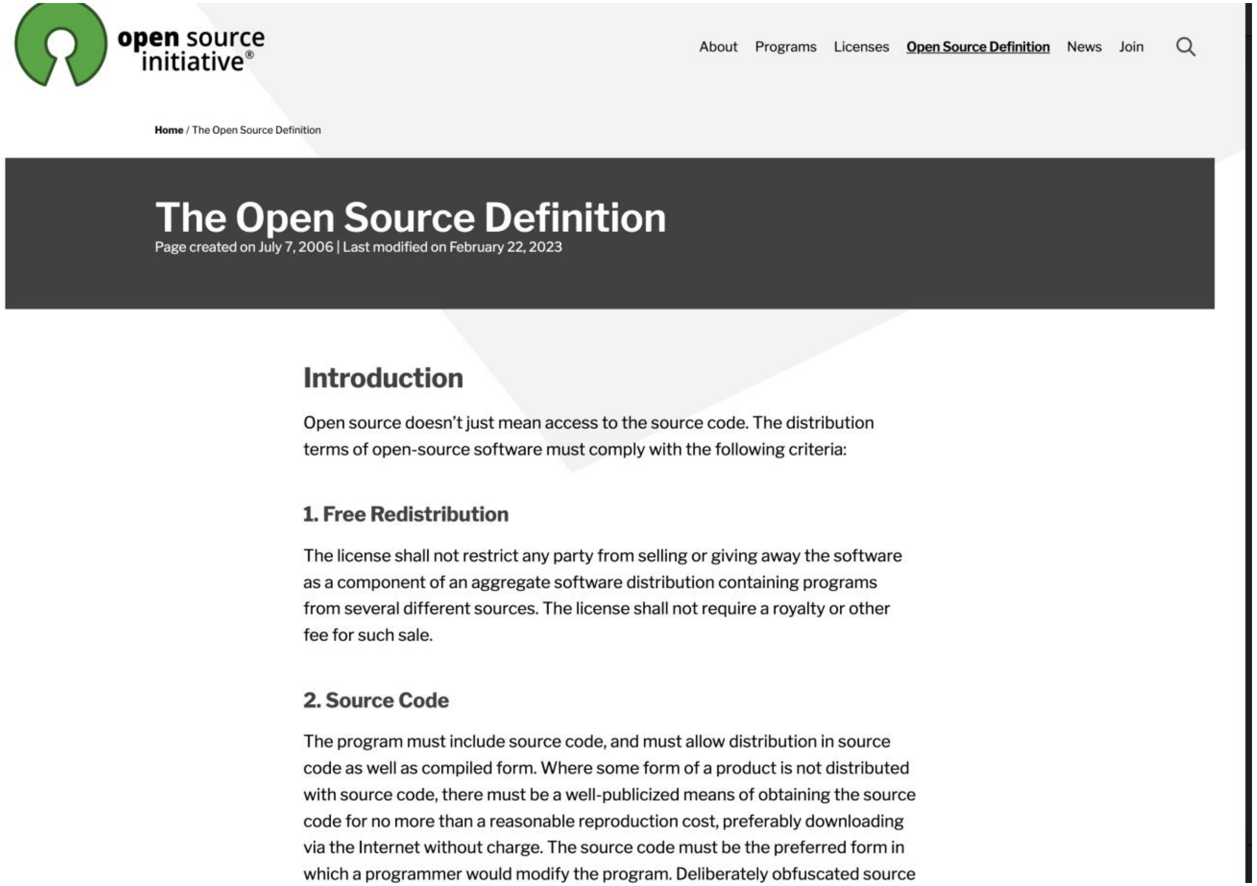
Guidance

Be open and use open source

Publish your code openly and use open source technology to improve transparency, flexibility and accountability.

Open Source Definition

- Open Source Criteria:
  - https://opensource.org/osd/



The Open Source Definition

Page created on July 7, 2006 | Last modified on February 22, 2023

Introduction

Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

1. Free Redistribution

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

2. Source Code

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost, preferably downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source

- Range of Code Adoption
  - Code adoption can happen at the level of a few lines of code, a method, a class, a library, a component, a tool, or a complete system.

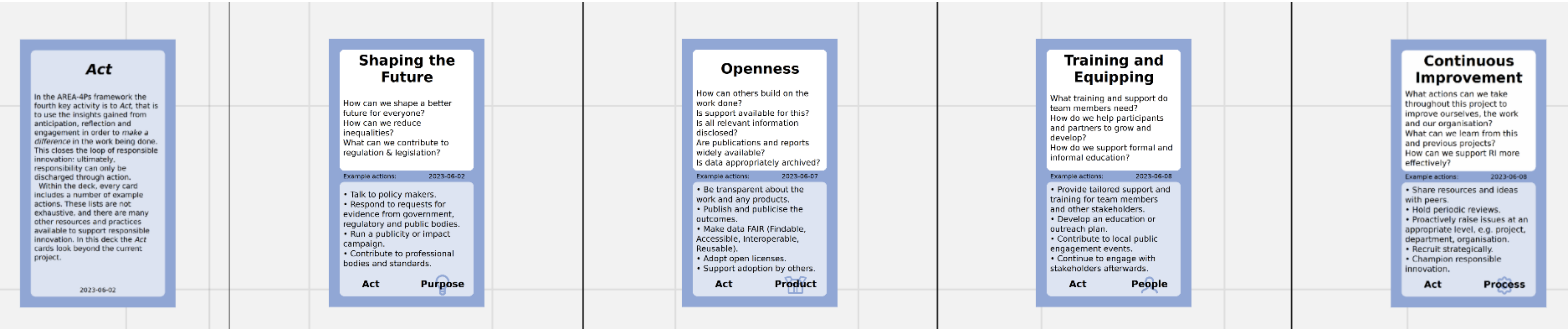
The Ethics of Open Source

</>

- Different points of view:
  - The ethics of free software (Dr Dobb's Journal 2000)
    - http://www.drdobbs.com/the-ethics-of-free-software/184414581
  - The ethics of open source software (Erfanian's Blog 2013)
    - http://www.ericerfanian.com/the-ethics-of-open-source-software/
  - Why open source software isn't as ethical as you think it is (Ethical Tech 2017)
    - https://words.werd.io/why-open-source-software-isnt-as-ethical-as-you-think-it-is-2e34d85c3b16

UoN - Responsible Research and Innovation

</>



Act

In the AREA-4Ps framework the fourth key activity is to *Act*, that is to use the insights gained from anticipation, reflection and engagement in order to *make a difference* in the work being done. This closes the loop of responsible innovation: ultimately, responsibility can only be discharged through action.

Within the deck, every card includes a number of example actions. These lists are not exhaustive, and there are many other resources and practices available to support responsible innovation. In this deck the *Act* cards look beyond the current project.

2023-06-02

Shaping the Future

How can we shape a better future for everyone?
How can we reduce inequalities?
What can we contribute to regulation & legislation?

Example actions: 2023-06-02

- Talk to policy makers.
- Respond to requests for evidence from government, regulatory and public bodies.
- Run a publicity or impact campaign.
- Contribute to professional bodies and standards.

Act Purpose

Openness

How can others build on the work done?
Is support available for this?
Is all relevant information disclosed?
Are publications and reports widely available?
Is data appropriately archived?

Example actions: 2023-06-07

- Be transparent about the work and any products.
- Publish and publicise the outcomes.
- Make data FAIR (Findable, Accessible, Interoperable, Reusable).
- Adopt open licenses.
- Support adoption by others.

Act Product

Training and Equipping

What training and support do team members need?
How do we help participants and partners to grow and develop?
How do we support formal and informal education?

Example actions: 2023-06-08

- Provide tailored support and training for team members and other stakeholders.
- Develop an education or outreach plan.
- Contribute to local public engagement events.
- Continue to engage with stakeholders afterwards.

Act People

Continuous Improvement

What actions can we take throughout this project to improve ourselves, the work and our organisation?
What can we learn from this and previous projects?
How can we support RI more effectively?

Example actions: 2023-06-08

- Share resources and ideas with peers.
- Hold periodic reviews.
- Proactively raise issues at an appropriate level, e.g. project, department, organisation.
- Recruit strategically.
- Champion responsible innovation.

Act Process

https://tas.ac.uk/responsible-research-innovation/rri-prompts-and-practice-cards/

licences

Software Licences

- I just want to give my software away! Why do I care about licences?

- Scenario 1
  - Bob installs my disk optimisation software. Great! But Bobs hard drive then catches fire and he loses an entire novel he is writing. Bad. Who's fault is it?

- Scenario 2
  - I write an awesome music sharing app. Great!
  - EvilCorp also like it, and they realise they can take it, close the source, and fill it with ads and sell it. Is this what I want?

Software Licences

- Software licences are there ...
  - to protect you as well as your code
  - to protect any future developers of the code

- We need rules in order to secure certain freedoms
  - What can be done with the code
  - Who can change it
  - Who can distribute it
  - Is there any warranty or disclaimer
  - ...

Software Licences

- Some common OSS licences (for more see https://opensource.org/licenses/ and https://en.wikipedia.org/wiki/Software_license)
  - Permissive license:
    - Subsequent users can produce 'closed source' versions and sell the software
  - CopyLeft license:
    - Any subsequent versions are left with the same rights e.g. source code must be supplied, and can be modified
- Examples
  - Apache Licence 2.0 > Permissive licence
  - BSD 3-Clause "New" or "Revised" licence > Permissive licence
  - GNU General Public Licence (GPL) > CopyLeft licence
  - MIT licence > Permissive licence
  - Mozilla Public Licence 2.0 > Permissive licence

</>

what next?

Using and Developing Your Skills

- What have you learnt during the COMP2013 module?
  - Much more experience with Java
  - Object-Oriented Design
  - GUI programming
  - Use of tools (e.g. Git/GitLab; IntelliJ; Maven/Gradle; …)
  - Modifying and adding to a sizeable existing project
- And with your group project you are learning to work as a team

- What now? You might want to:
  1. Initiate your own open source project
  2. Contribute to an existing project

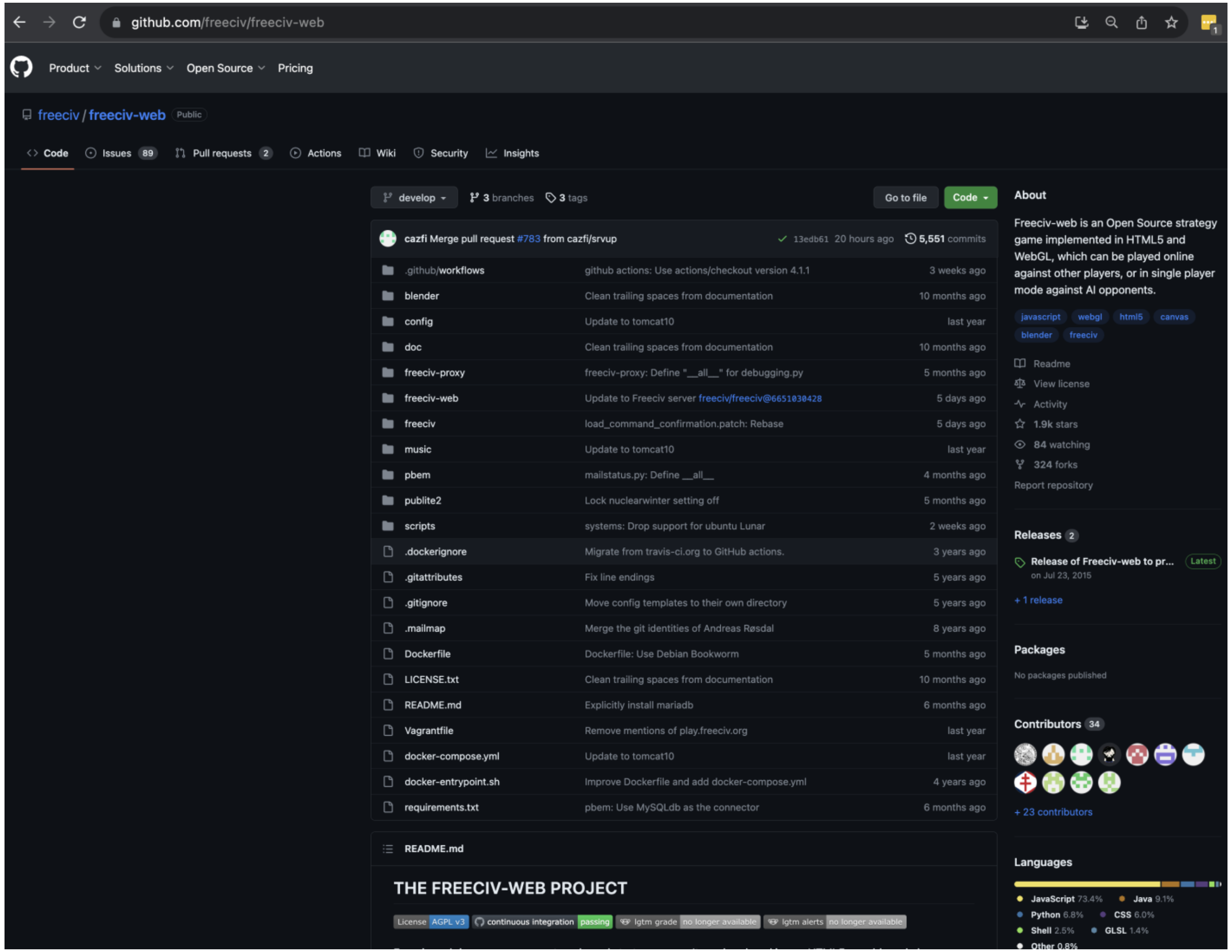
Initiate Your Own Project

</>

- How?
  - Identify a need
  - Discuss the idea
  - Does it already exist?
  - Talk to your lecturers / peers
  - Code Development starts
  - Create website and "announce"
  - Community evolves
  - Functionalities added

- Funding?
  - Donation, voluntary, crowd-funded

Contribute to an Existing Project



cazfi Merge pull request #783 from cazfi/srvup ✓ 13edb61 20 hours ago 5,551 commits

.github/workflows	github actions: Use actions/checkout version 4.1.1	3 weeks ago
blender	Clean trailing spaces from documentation	10 months ago
config	Update to tomcat10	last year
doc	Clean trailing spaces from documentation	10 months ago
freeciv-proxy	freeciv-proxy: Define "__all__" for debugging.py	5 months ago
freeciv-web	Update to Freeciv server freeciv/freeciv@6651030428	5 days ago
freeciv	load_command_confirmation.patch: Rebase	5 days ago
music	Update to tomcat10	last year
pbem	mailstatus.py: Define __all__	4 months ago
publite2	Lock nuclearwinter setting off	5 months ago
scripts	systems: Drop support for ubuntu Lunar	2 weeks ago
.dockerignore	Migrate from travis-ci.org to GitHub actions.	3 years ago
.gitattributes	Fix line endings	5 years ago
.gitignore	Move config templates to their own directory	5 years ago
.mailmap	Merge the git identities of Andreas Røsdal	8 years ago
Dockerfile	Dockerfile: Use Debian Bookworm	5 months ago
LICENSE.txt	Clean trailing spaces from documentation	10 months ago
README.md	Explicitly install mariadb	6 months ago
Vagrantfile	Remove mentions of play.freeciv.org	last year
docker-compose.yml	Update to tomcat10	last year
docker-entrypoint.sh	Improve Dockerfile and add docker-compose.yml	4 years ago
requirements.txt	pbem: Use MySQLdb as the connector	6 months ago

README.md

THE FREECIV-WEB PROJECT

License AGPL v3 | continuous integration passing | lgtm grade no longer available | lgtm alerts no longer available

About

Freeciv-web is an Open Source strategy game implemented in HTML5 and WebGL, which can be played online against other players, or in single player mode against AI opponents.

javascript webgl html5 canvas blender freeciv

Readme
View license
Activity
1.9k stars
84 watching
324 forks
Report repository

Releases 2

Release of Freeciv-web to pr... Latest
on Jul 23, 2015

+ 1 release

Packages

No packages published

Contributors 34

+ 23 contributors

Languages

● JavaScript 73.4% ● Java 9.1%
● Python 6.8% ● CSS 6.0%
● Shell 2.5% ● GLSL 1.4%
● Other 0.8%

https://github.com/freeciv/freeciv-web
https://github.com/freeciv/freeciv-web/issues

Contribute to an Existing Project

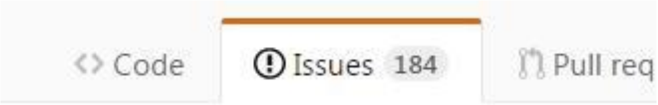
- How to get involved
  - Look at the README file
    - Should explain the purpose of the projects, direction of development, etc.
    - Look out for sections on How To Contribute
  - Fork the project
    - Create a branch?
  - Check for any rules on how contributors should work
    - Respect the rules of the project

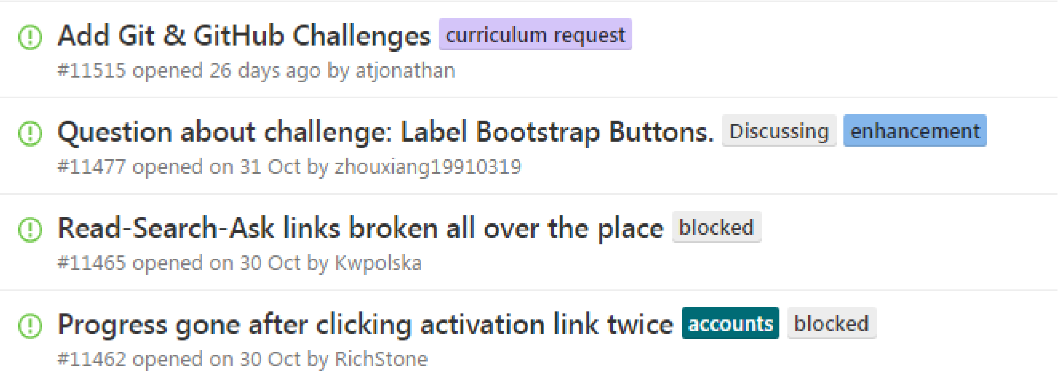
Contribute to an Existing Project

- How to get involved (cont.)
  - Join a development chat or forum to see how development is organised
  - Start to talk to the developers if you think you can help
  - Learn how to use merge/pull requests
  - Write a test to show you have fixed a bug
  - Have the right attitude!
    - Polite; patient

Contribute to an Existing Project

- Looking for open issues
  - Look under Issues for a project
  - Projects can assign labels
  - Some target new developers



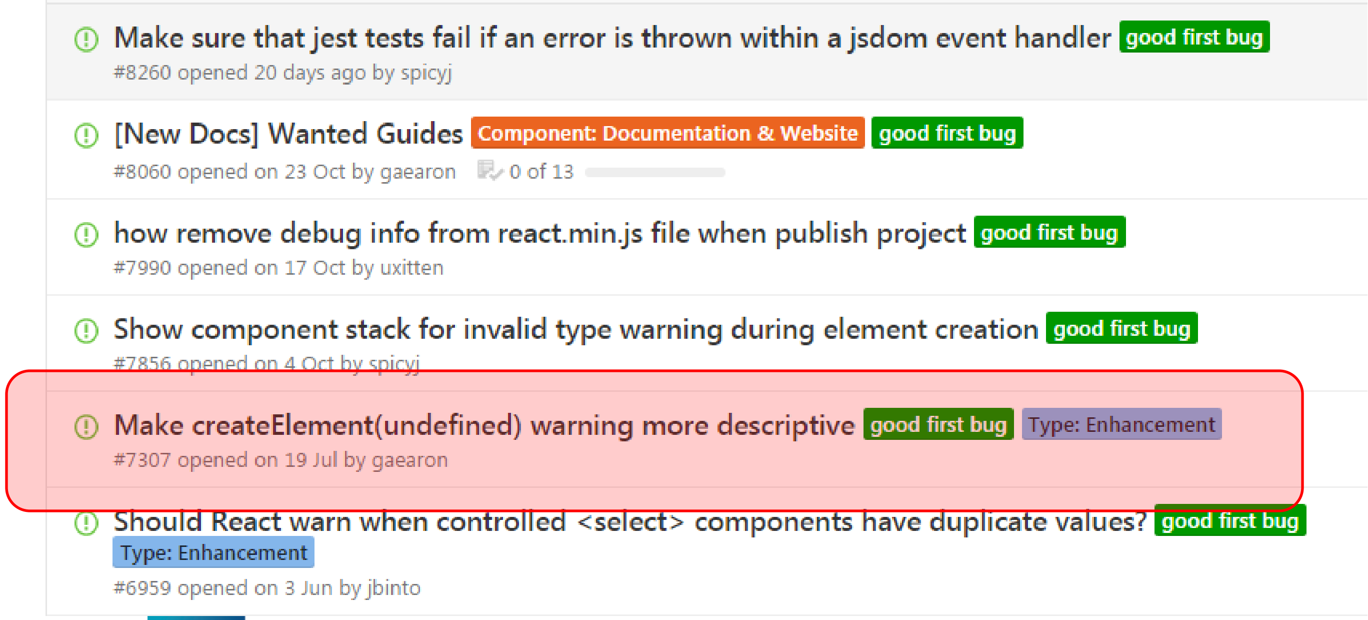


Add Git & GitHub Challenges curriculum request
#11515 opened 26 days ago by atjonathan

Question about challenge: Label Bootstrap Buttons. Discussing enhancement
#11477 opened on 31 Oct by zhouxiang19910319

Read-Search-Ask links broken all over the place blocked
#11465 opened on 30 Oct by Kwpolska

Progress gone after clicking activation link twice accounts blocked
#11462 opened on 30 Oct by RichStone



Make sure that jest tests fail if an error is thrown within a jsdom event handler good first bug
#8260 opened 20 days ago by spicyj

[New Docs] Wanted Guides Component: Documentation & Website good first bug
#8060 opened on 23 Oct by gaearon 0 of 13

how remove debug info from react.min.js file when publish project good first bug
#7990 opened on 17 Oct by uxitten

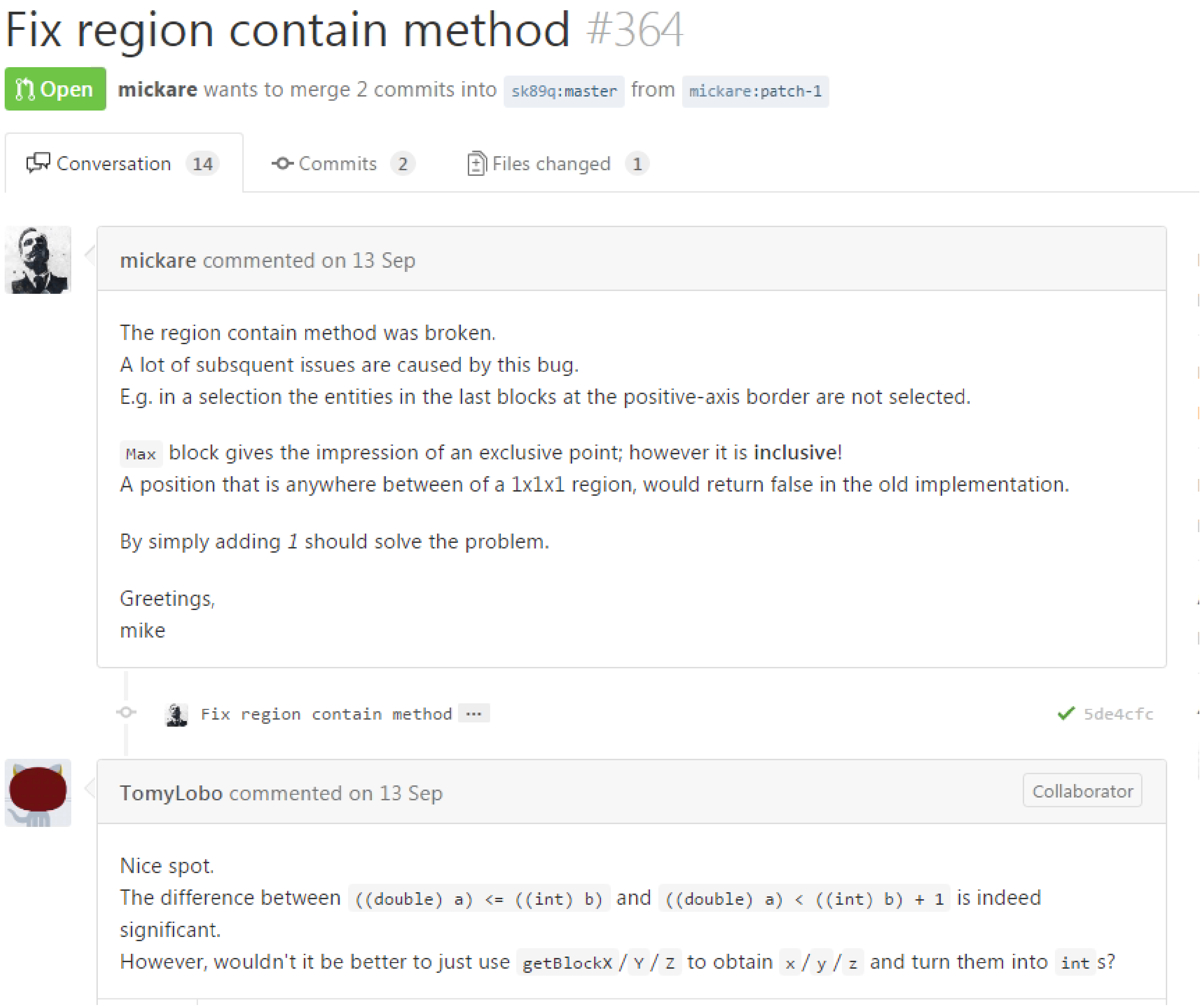
Show component stack for invalid type warning during element creation good first bug
#7856 opened on 4 Oct by spicyj

Make createElement(undefined) warning more descriptive good first bug Type: Enhancement
#7307 opened on 19 Jul by gaearon

Should React warn when controlled <select> components have duplicate values? good first bug Type: Enhancement
#6959 opened on 3 Jun by jbinto

Contribute to an Existing Project

- Communication is informal
  - Community communications
  - Threaded discussion forums
  - Email (list servers)
  - Newsgroups
  - Messaging/chat
  - Community digests
  - Social networks



Fix region contain method #364

Open **mickare** wants to merge 2 commits into `sk89q:master` from `mickare:patch-1`

mickare commented on 13 Sep

The region contain method was broken.
A lot of subsquent issues are caused by this bug.
E.g. in a selection the entities in the last blocks at the positive-axis border are not selected.

`Max` block gives the impression of an exclusive point; however it is **inclusive**!
A position that is anywhere between of a 1x1x1 region, would return false in the old implementation.

By simply adding *1* should solve the problem.

Greetings,
mike

`Fix region contain method` … ✓ 5de4cfc

TomyLobo commented on 13 Sep Collaborator

Nice spot.
The difference between `((double) a) <= ((int) b)` and `((double) a) < ((int) b) + 1` is indeed significant.
However, wouldn't it be better to just use `getBlockX` / `Y` / `Z` to obtain `x` / `y` / `z` and turn them into `int` s?

Challenges of Collaborative OSS Development

</>

- Product structure and comprehension
  - Who understands the 'whole system'?
- Effective ways of incorporating requirements of non-developer users?
- With larger scales, will coordination needs force adoption of "commercial" development techniques?
- How to collaborate on "big" features?
- How to respond to unanticipated events? (Funding etc.)

and finally …

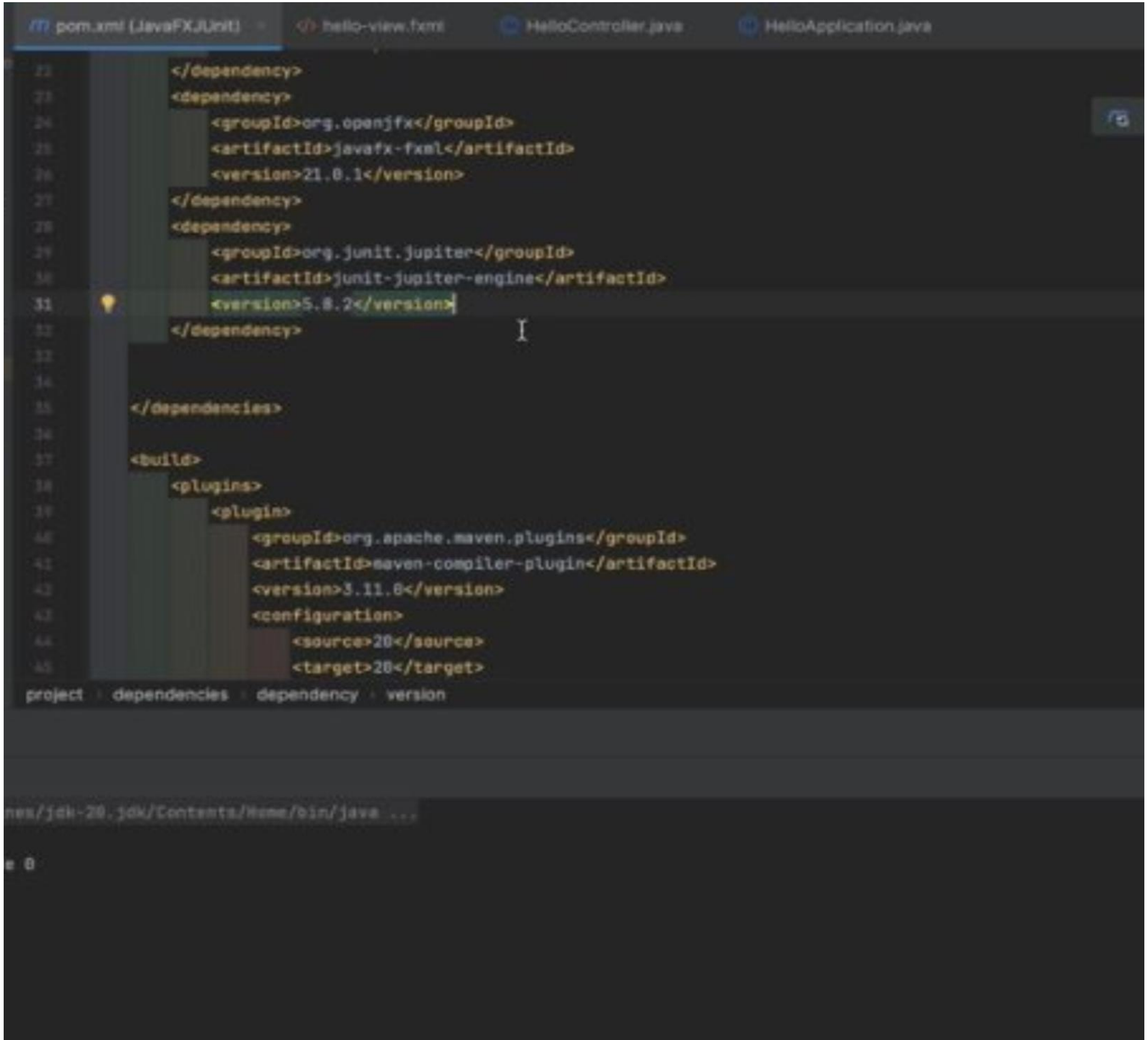
Reading for First Timers: http://www.firsttimersonly.com/

GitHub for Beginner's Guides and Help: https://github.com/btford/participating-in-open-source/

Acknowledgements

Thanks to Peer-Olaf Siebers and Robert Laramee for the lecture materials

JUnit refresher



```
        </dependency>
        <dependency>
            <groupId>org.openjfx</groupId>
            <artifactId>javafx-fxml</artifactId>
            <version>21.0.1</version>
        </dependency>
        <dependency>
            <groupId>org.junit.jupiter</groupId>
            <artifactId>junit-jupiter-engine</artifactId>
            <version>5.8.2</version>
        </dependency>


    </dependencies>

    <build>
        <plugins>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-compiler-plugin</artifactId>
                <version>3.11.0</version>
                <configuration>
                    <source>20</source>
                    <target>20</target>
```

```
nes/jdk-20.jdk/Contents/Home/bin/java ...

e 0
```

[JUnit Test - HD 1080p.mov]