



## Your Project

Software Engineering II  
WS 2017/18

Arian Treffer  
arian.treffer@hpi.de

Prof. Plattner, Dr. Uflacker  
Enterprise Platform and Integration Concepts group

# SWT2 Schedule



## PROJECT

- **November 10: Kick-Off**
- Nov 13 – Nov 17: Begin of Sprint 1
  
- Exact dates for meetings should be **negotiated** with your tutor
  - Enter dates and rooms in public calendar
- Sprint reviews and plannings for next sprint can be merged
- PO should roughly know what the team has done **before the review!**
- **On demand:** User Research with Customer

# Let's get started



- POs
  - Extract Requirements + create user stories (Github Tickets)
  - **Prepare** Sprint planning
- Teams
  - Prepare working environment
  - Clone repository, try to get application working, understand architecture
  - **Play around**
  - Find a regular timeslot for your meetings
- There will be a **tutorium in this space after the second lecture (15:00+)**
  - Ask any (Ruby) question
  - Setup project and development environment together



# Project Infrastructure

# Communication Infrastructure



- Email lists
  - If you want: separate lists for each team (*lists.myhpi.de*)
  - Keep your teammates in the loop, add teaching team
  - Rules and filters help organizing your inbox
- <https://swt2-2017.slack.com>
- Ticket system for overview and feedback about current tasks and progress
- Wiki for lean and globally **accessible documentation**
- Telephone and personal contact for direct communication
- ... be **creative!**  
(but let us know, we are interested in learning what might be useful in the future)

# Time Management



## Google Calendar

- Advantages:
  - Available Everywhere
  - Easy Integration with Outlook & iCal
- Overview of team appointments
- Access granted by our tutors

# Application Lifecycle Management



The Swiss army knife of software development

- Integrating tools for most common activities in **one place**
- Wiki, Bug Tracking, Time Management, Project Analytics, Discussions, ...
- Examples:
  - MS Team Foundation Server
  - Codebeamer
  - Trac
  - Redmine, Plan.io (SaaS based on Redmine)
  - **Github**

# Github Project



■ <https://github.com/hpi-sw22/sport-portal/>

The screenshot shows a web browser displaying the GitHub repository page for 'sport-portal'. The browser's address bar shows the URL 'https://github.com/hpi-sw22/sport-portal'. The repository's file list is visible, including 'LICENSE', 'Profile', 'README.md', 'Rakefile', 'Vagrantfile', 'config.ru', and 'generates.txt'. The 'README.md' file is selected and its content is displayed below. The README includes a title 'sport-portal', a description 'A Ruby on Rails app to manage sport matches', a table of deployment status for different branches, and a 'Local Setup' section with instructions for cloning the repository and installing dependencies.

Branch	Travis CI	Code Coverage	Code Analysis	Heroku Deploy	Errbit
dev (default)	build passing	coverage 100%	maintainability ▲	heroku deployed [link]	[link]
master	build passing	coverage 100%		heroku deployed [link]	[link]

### Local Setup

- Clone this repository
- cat .ruby-version && echo \$(ruby --version) See if locally installed ruby version matches the one specified in the .ruby\_version file.
- If the ruby version is different: Install the required version using rbenv (recommended) or RVM
- gem install bundler Install bundler for managing Ruby gems
- bundle install Install the required Ruby gem dependencies defined in the project's Gemfile
- rails db:create db:migrate db:seed Setup database, run migrations, seed the database with default



# Version Control System



Repository to store software configuration items

Features:

- Versioning
- Dealing with variants: **branches**
- Access control
  - Authentication, authorization
  - Locking
  - **Concurrent** development
- Reporting
  - How many versions, variants, changes, persons
  - History of changes

# Continuous Integration



How to check **continuously** that your software works?

Continuous Integration!

- **Connected** to version control
- Customizable run scripts
- Ideally covering all development branches
- Examples:
  - CruiseControl
  - Anthill
  - Jenkins/Hudson
  - **Travis-CI**

# Travis CI



- <https://travis-ci.org/hpi-swt2/sport-portal>

Travis CI About Us Blog Status Help [Sign in with GitHub](#)

Help make Open Source a better place and start building better software today!

## hpi-swt2 / sport-portal build passing

[Current](#) [Branches](#) [Build History](#) [Pull Requests](#) [More options](#)

### Default Branch

✓ dev 38 builds	# 124 passed 27 minutes ago	3eb1797 GitHub	✓	✓	✓	✓	✓
--------------------	--------------------------------	-------------------	---	---	---	---	---

### Active Branches

✓ master 80 builds	# 120 passed about 5 hours ago	3f8e890 Christoph Matthies	✓	⊘	✓	✓	✓
-----------------------	-----------------------------------	-------------------------------	---	---	---	---	---

### Inactive Branches

✓ revert-15-dev 1 build	# 118 passed about 5 hours ago	e403454 GitHub	✓				
----------------------------	-----------------------------------	-------------------	---	--	--	--	--

# Application Deployment

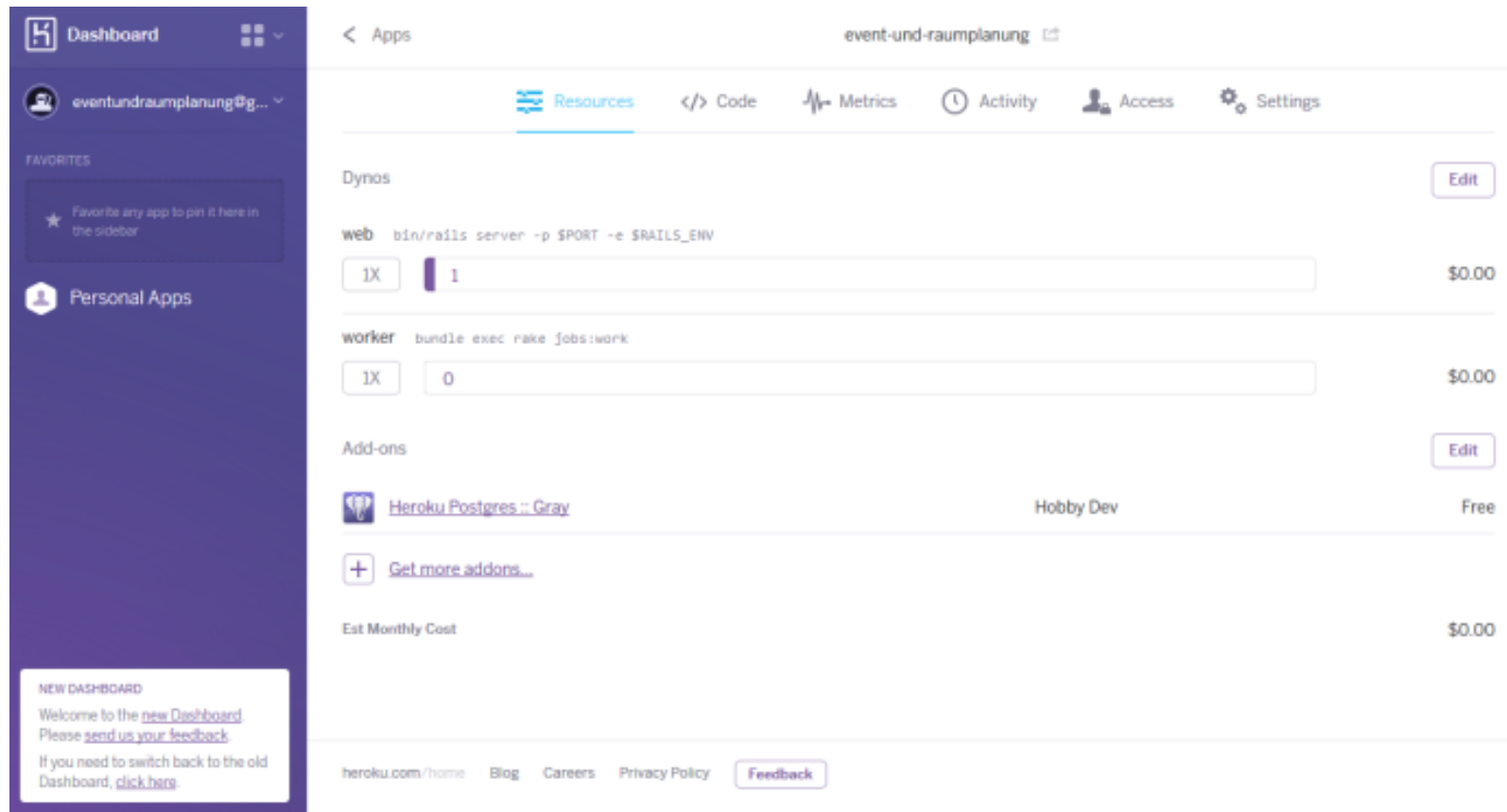


## How to have a **running version** available at all times?

Deploy your app!

- Simple solution: local deployment
- Dedicated Servers, Infrastructure-as-a-Service, Platform-as-a-Service
  - more about that in a separate lecture
- We use **Heroku** (PaaS)
- Deployment is **automatically** triggered by successful Travis-CI build
- Errors that occur in production sent to a central system: Errbit

■ <http://heroku.com/>



The screenshot shows the Heroku dashboard for an application named "event-und-raumplanung". The interface includes a sidebar with navigation options like "Dashboard", "Personal Apps", and "Favorites". The main content area displays the application's configuration, including Dynos (web and worker) and Add-ons (Heroku Postgres). The estimated monthly cost is shown as \$0.00.

Dynos	Configuration	Cost
web	bin/rails server -p \$PORT -e \$RAILS_ENV	\$0.00
worker	bundle exec rake jobs:work	\$0.00

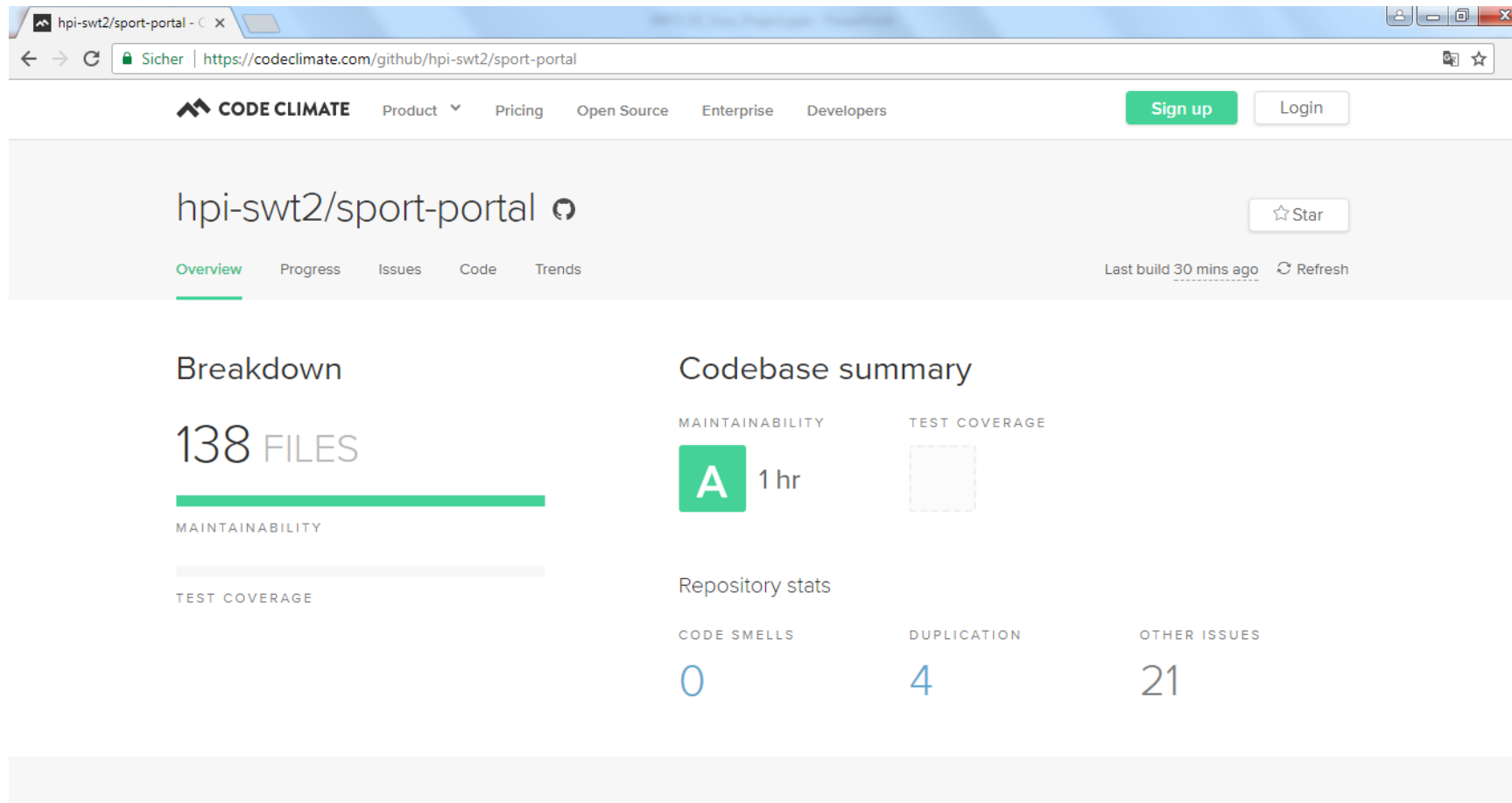
Add-ons	Configuration	Cost
Heroku Postgres :: Gray	Hobby Dev	Free

Est Monthly Cost: \$0.00

How to ensure that your software adheres to certain **quality standards** (complexity, test coverage, etc.)?

- Self-control
- Code Reviews (future lecture)
- **Automatic** checking
  - CodeClimate
  - SimpleCov
    - runs automatically during each test run
    - coverage/index.html in your application folder

- <https://codeclimate.com/github/hpi-sw22/sport-portal>



The screenshot shows the CodeClimate dashboard for the repository hpi-sw22/sport-portal. The page includes a navigation bar with links for Product, Pricing, Open Source, Enterprise, and Developers, along with Sign up and Login buttons. The repository name is displayed with a Star button and a refresh button. The main content area is divided into two sections: Breakdown and Codebase summary. The Breakdown section shows 138 files, a green bar for Maintainability, and a grey bar for Test Coverage. The Codebase summary section displays a green 'A' grade with a 1 hr estimate, a dashed box for Test Coverage, and Repository stats showing 0 Code Smells, 4 Duplications, and 21 Other Issues.

Category	Value
Files	138
Maintainability	A (1 hr)
Test Coverage	(Dashed box)
Code Smells	0
Duplications	4
Other Issues	21

All Files (100.0%)   Controllers (100.0%)   Models (100.0%)   Mailers (100.0%)   Helpers (100.0%)   Libraries (100.0%)   Plugins (100.0%)

## All Files (100.0% covered at 1.35 hits/line)

6 files in total. 41 relevant lines. 41 lines covered and 0 lines missed

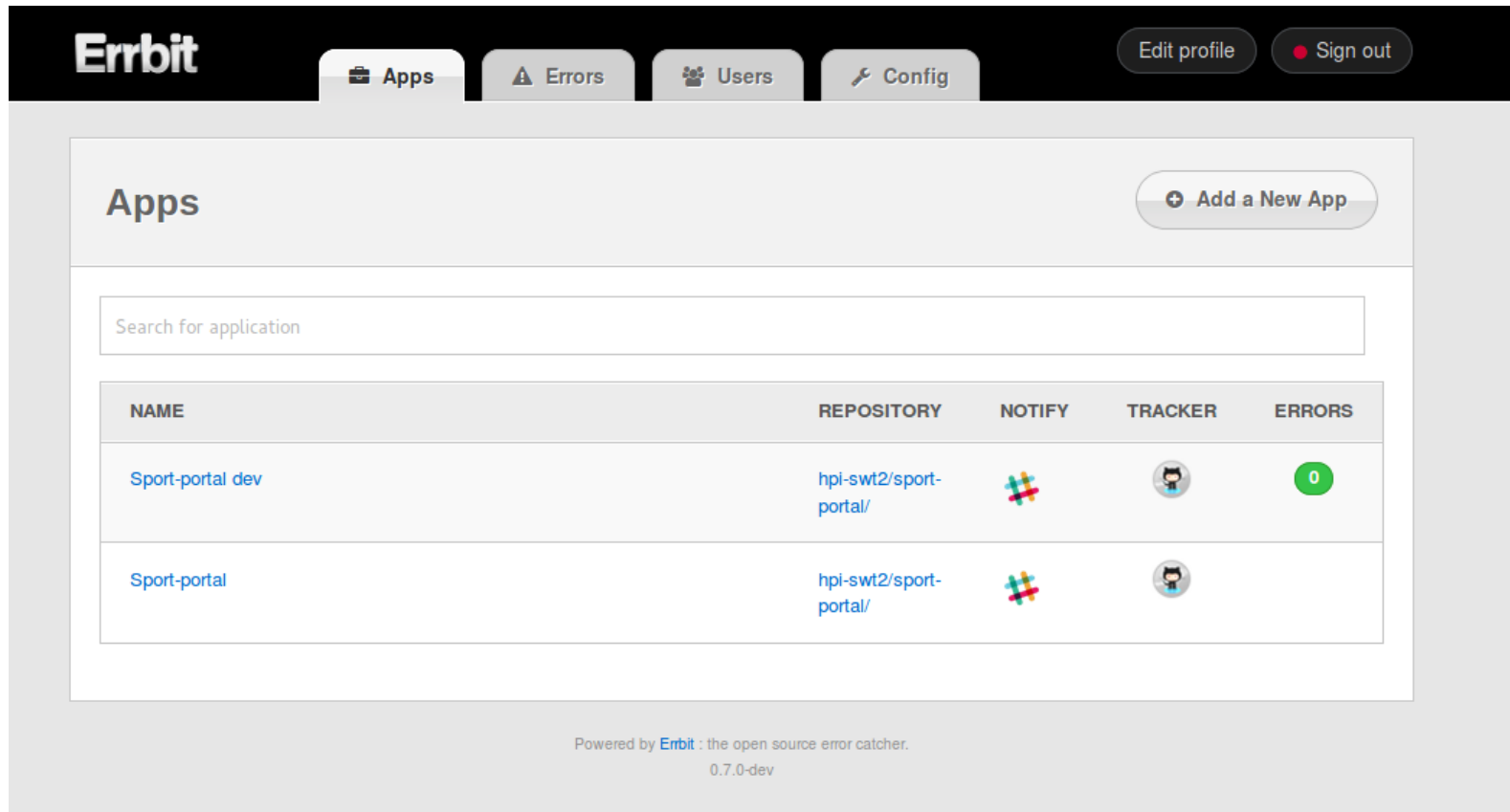
Search:

File	% covered	Lines	Relevant Lines	Lines covered
app/controllers/application_controller.rb	100.0 %	5	2	2
app/controllers/job_offers_controller.rb	100.0 %	77	34	34
app/helpers/application_helper.rb	100.0 %	2	1	1
app/helpers/job_offers_helper.rb	100.0 %	2	1	1
app/models/job_offer.rb	100.0 %	2	1	1
app/models/user.rb	100.0 %	7	2	2





Showing 1 to 6 of 6 entries



- <https://swt2-errbit-2017.herokuapp.com/>



The screenshot shows the Errbit web interface. At the top, there is a navigation bar with the Errbit logo and several menu items: Apps, Errors, Users, and Config. On the right side of the navigation bar, there are buttons for 'Edit profile' and 'Sign out'. Below the navigation bar, the main content area is titled 'Apps' and features a search bar labeled 'Search for application'. To the right of the search bar is a button labeled '+ Add a New App'. Below the search bar is a table with the following columns: NAME, REPOSITORY, NOTIFY, TRACKER, and ERRORS. The table contains two rows of application data.

NAME	REPOSITORY	NOTIFY	TRACKER	ERRORS
Sport-portal dev	hpi-swt2/sport-portal/			0
Sport-portal	hpi-swt2/sport-portal/			

Powered by [Errbit](#) : the open source error catcher.  
0.7.0-dev

# Your Project



1. Schedule
2. Calendar
3. Slack
4. Github
5. Travis CI
6. Heroku
7. CodeClimate
8. Errbit

# Next Week



Week 5 (Nov 13 – Nov 17)

- PO presents team vision
- Estimation
- Begin 1<sup>st</sup> sprint
- Planning

# Image Credits



- "ST vs Gloucester - Match - 23" by PierreSelim - Own work. Licensed under Creative Commons Attribution-Share Alike 3.0 via Wikimedia Commons - [http://commons.wikimedia.org/wiki/File:ST\\_vs\\_Gloucester\\_-\\_Match\\_-\\_23.JPG#mediaviewer/File:ST\\_vs\\_Gloucester\\_-\\_Match\\_-\\_23.JPG](http://commons.wikimedia.org/wiki/File:ST_vs_Gloucester_-_Match_-_23.JPG#mediaviewer/File:ST_vs_Gloucester_-_Match_-_23.JPG)
- "Scrum process" by Lakeworks - Own work. Licensed under Creative Commons Attribution-Share Alike 3.0-2.5-2.0-1.0 via Wikimedia Commons - [http://commons.wikimedia.org/wiki/File:Scrum\\_process.svg#mediaviewer/File:Scrum\\_process.svg](http://commons.wikimedia.org/wiki/File:Scrum_process.svg#mediaviewer/File:Scrum_process.svg)
- „Wien - Seestadt, SW-Areal 2013 (2)“ von Bwag - Eigenes Werk. Lizenziert unter Creative Commons Attribution-Share Alike 3.0-at über Wikimedia Commons - [http://commons.wikimedia.org/wiki/File:Wien\\_-\\_Seestadt,\\_SW-Areal\\_2013\\_\(2\).JPG#mediaviewer/File:Wien\\_-\\_Seestadt,\\_SW-Areal\\_2013\\_\(2\).JPG](http://commons.wikimedia.org/wiki/File:Wien_-_Seestadt,_SW-Areal_2013_(2).JPG#mediaviewer/File:Wien_-_Seestadt,_SW-Areal_2013_(2).JPG)
- "Utility pole in Curitiba" by Leonardo.stabile - Own work. Licensed under Public domain via Wikimedia Commons - [http://commons.wikimedia.org/wiki/File:Utility\\_pole\\_in\\_Curitiba.JPG#mediaviewer/File:Utility\\_pole\\_in\\_Curitiba.JPG](http://commons.wikimedia.org/wiki/File:Utility_pole_in_Curitiba.JPG#mediaviewer/File:Utility_pole_in_Curitiba.JPG)