

## Assignment Brief Form

### Course information from DRPS

<b>Course acronym</b>	<b>ACP</b>
<b>Course code</b>	<b>INFR11245 / INFR11249</b>
<b>Credits</b>	10
<b>Course Organiser(s)</b>	Michael Glienecke
<b>Learning Outcomes</b> <i>Paste in from DRPS</i>	<p>On completion of this course, the student will be able to:</p> <ol style="list-style-type: none"> <li>1. implement the basic building blocks of cloud programming like containerization, micro-services, event-processing and cloud data storage using standard service offerings by the main cloud providers in combination with several communication protocols</li> <li>2. compare / contrast the differences between architecture styles like IaaS, PaaS, CaaS, SaaS, etc</li> <li>3. summarize cloud service offerings by the main providers and evaluate the differences</li> <li>4. contrast different contemporary cloud runtime environments</li> <li>5. differentiate the need and structure of Continuous Integration / Continuous deployment (CI/CD)-chains in cloud programming</li> </ol>

### Mapping of Learning Outcomes onto Assessment

Please indicate which Learning Outcomes are assessed by which assessment components by entering a “+” in the cells of the table below.

	LO1	LO2	LO3	LO4	LO5
CW1	++		+		
CW2	+++		+++		+
CW3 (student choice)	++				
CW4 (video presentation)		++	+	++	+

## Brief for each coursework

### CW1

<b>Assignment name</b> <i>This name should correspond to the Coursework Name in PIP</i>	Programming Assignment
<b>Task overview</b>	The student must implement a rudimentary Java REST-service with some 4-5 pre-defined endpoints which are re-used in CW2. Simple logic, mostly to check for setup, REST-basics and simple testing
<b>Assessment criteria</b>	Each task carries points. For minor deviations (e.g. slightly incorrect results) a penalty is taken; otherwise, 0 points for the given task
<b>Submission instructions</b>	Docker-image plus all source-codes as Learn submission
<b>Guidance on size of submission</b>	N/A
<b>Penalties for overlong submissions</b>	N/A
<b>Feedback procedure</b>	E-Mail with individual auto-marker results (complete transcript), discussion in class about the most common problems and Q&A / individual sessions if needed
<b>Support arrangements</b>	Tutorials, Q&A, TA, piazza
<b>Marking and moderation procedure</b>	Docker-image is auto-marked producing a result record with points (uploaded to Learn) The auto-marker produces detailed information per task and highlights problems  For moderation students reach out to the CO and then discuss the issues.

## CW2

<b>Assignment name</b> <i>This name should correspond to the Coursework Name in PIP</i>	Programming Assignment
<b>Task overview</b>	Like CW1, just more elaborate (5-6 endpoints and more complex implementation logic).
<b>Assessment criteria</b>	Each task carries points. For minor deviations (e.g. slightly incorrect results) a penalty is taken; otherwise, 0 points for the given task
<b>Submission instructions</b>	Docker-image plus all source-codes as Learn submission
<b>Guidance on size of submission</b>	N/A
<b>Penalties for overlong submissions</b>	N/A
<b>Feedback procedure</b>	E-Mail with individual auto-marker results (complete transcript), discussion in class about the most common problems and Q&A / individual sessions if needed
<b>Support arrangements</b>	Tutorials, Q&A, TA, piazza
<b>Marking and moderation procedure</b>	<p>Docker-image is auto-marked producing a result record with points (uploaded to Learn)</p> <p>The auto-marker produces detailed information per task and highlights problems</p> <p>For moderation students reach out to the CO and then discuss the issues.</p>

## CW3

<b>Assignment name</b> <i>This name should correspond to the Coursework Name in PIP</i>	Student choice implementation
<b>Task overview</b>	<p>The student can implement anything in relation to the contents of ACP in either Java, Python, JavaScript, react, Go or Rust.</p> <p>This could be a UI, some complex pre- / post-processing system, some data pipeline, etc. MUST (!) utilize a free service from any cloud provider</p> <p>Student's creativity is the limit</p>
<b>Assessment criteria</b>	<p>The main criterion for marking is the PDF, the sources only serve as evidence or investigation point.</p> <ul style="list-style-type: none"> <li>• Innovation / Idea / Benefit (4)</li> <li>• Execution / Implementation (8)</li> <li>• Completeness (4)</li> <li>• Style (4)</li> </ul>
<b>Submission instructions</b>	<p>The student must submit all sources + necessary docker images plus a description PDF which explains the implementation: Why, for whom, what problem shall be solved / mitigated, details of implementation.</p> <p>AI is allowed for planning</p>
<b>Guidance on size of submission</b>	<p>ZIP-file + PDF in Learn</p> <p>PDF not more than 1000 words / 4 pages with diagrams as guidance.</p>
<b>Penalties for overlong submissions</b>	<p>More than 50% excess will incur a 25% deduction of marks</p>
<b>Feedback procedure</b>	<p>Feedback in Learn and based on the rubrics</p>
<b>Support arrangements</b>	<p>Tutorials, piazza, Q&amp;A, TA, direct contact</p>
<b>Marking and moderation procedure</b>	<p>Marked by 3 markers in parallel to avoid bias (medium is taken)</p>

## CW4

<b>Assignment name</b> <i>This name should correspond to the Coursework Name in PIP</i>	Video presentation of the student implementation plus additional topics
<b>Task overview</b>	<p>The student provides a 8-11 min video where the implementation is presented alongside background, motivation, target audience.</p> <p>In addition, several predefined questions regarding learning outcome 3 / 4 / 5 must be presented and set in context to the chosen implementation</p>
<b>Assessment criteria</b>	<ul style="list-style-type: none"> <li>- Background elaboration / target audience (10%)</li> <li>- Problem statement (10%)</li> <li>- Explanation of the student's choice (20%)</li> <li>- Presentation (20%)</li> <li>- Pre-defined questions (20%)</li> <li>- Elaboration of relation to student's implementation (20%)</li> </ul>
<b>Submission instructions</b>	<p>Upload to Learn as a video.</p> <p>If the video is too long -&gt; 10% deduction for every minute over 11;</p>
<b>Guidance on size of submission</b>	N/A (8 - 11 min)
<b>Penalties for overlong submissions</b>	10% deduction for every minute over 11;
<b>Feedback procedure</b>	In Learn based on the Rubrics
<b>Support arrangements</b>	Tutorials, piazza, Q&A, TA, direct contact
<b>Marking and moderation procedure</b>	Marked by 2 persons (usually CO + TA) to avoid bias (arithmetic mean is taken)