ACG Course   
proposal

|  |  |
| --- | --- |
| Date | 03/5/2019 |

Contact information

Subject to the acceptance of your course proposal by ACG Content Management and your acceptance of the A Cloud Guru contractual terms, the contact information listed here will be used to populate the contract. By listing your company name, you will be entering into the agreement as a company. If you would like to contract as an individual, do not include any information in the company name section.

|  |  |
| --- | --- |
| Instructor name | Wolfgang Unger |
| Company name*(if applicable)* | Currently NTT, but doing this not as employee but as freelancer. |
| Address  (if contracting as a company, please list your company address here) |  |
| Email | wolfgang.unger@gmx.de |
| Contact number  (Please include country and region codes) | +4917664182900 |

Course summary

|  |  |
| --- | --- |
| Course title  *(Provide a clear and engaging title for your course – so student’s will know exactly what it’s about, and they’ll want to know more!)* | Deep Dive on AWS DevOps CI/CD Tooling |
| Elevator pitch  *(Sum up your course in one line. The snappier the better!)* | DevOps concepts, best practices and how to use AWS CodeCommit, CodeBuild, CodeDeploy and CodePipeline. |
| Shortsummary  (Provide a bullet point list of the main tasks/topics that will be covered in the course) | In this course, you’ll learn:   * How to use CodeCommit and integrate it in popular IDEs, Triggers, Pull Requests and Branches. * How to use CodeBuild to build your projects, test your projects, input sources, output artifacts, monitoring and env. variables. * How to use CodeDeploy to deploy your artifacts. * How to use CodePipeline to put it all together, create multi-staged pipelines, hands-on experience on many different example pipelines. |
| Fulldescription  (Provide a full description, or blurb, for your course. This will be used as the basis for the course description on the platform. Please aim for about 100 to 150 words. Don’t worry too much about finessing this as our marketing department will help with polishing and optimising the copy.) | DevOps is a new term combing 2 major areas in the software development process. These are Development and Operations.  DevOps has become more and more important within the last few years, and today a profound ‘know how’ is almost mandatory in the agile development process.  Learn the concepts, fundamentals and best practices of DevOps using the AWS services CodeCommit, CodeBuild, CodeDeploy and CodePipeline with hands on examples!  Basic software development experience is expected, and you should be able to code simple applications. |
| Learningoutcomes  (Provide 5 to 6 learning outcomes from the course. These LOs should be achievable and measurable. Will students ‘build’ something, ‘analyse’ something’, ‘create’ something etc.  Your learning outcomes should link clearly to your course description and the structure of your course.) | After completing this course, you will be able to:   1. Use and maintain CodeCommit as a version control system and integrate in common IDEs. 2. Create and maintain CodeBuild Projects and then run, monitor and troubleshoot these projects. 3. Create and maintain CodeDeploy jobs to automatically deploy your artifacts. 4. Create and maintain Build Pipelines for different use-cases and languages, multi-staged, trigger these pipelines with web-hooks and cloud watch events. |
| Skill level  (Introductory level = non-technical; Beginner level = basic exposure to the platform, technology or service.) | **Introductory**   **Beginner**  **Intermediate**   **Advanced**  Not sure, advanced should be selected. Depends on the definition of advanced …. Please make your choice |
| Audience  (List the audience that the course is designed for) | * Developers who want to expand their ‘know how’ with AWS DevOps skills and concepts of modern, agile software development processes. * Operations engineers that want to understand how to migrate Operations and DevOps to AWS. * Technical managers whose teams are moving to cloud technology and want to understand the concepts of DevOps on AWS. * Other people involved with technical teams, including quality assurance, operations, product management, and project management. |
| Prerequisites  (List any prerequisites in terms of knowledge and skills that the student needs to have to be able to engage with the course material.) | At a minimum you will need:   * Software development basics and experience. * DevOps basics would be an advantage. * GIT basics or experience would be an advantage. * Software development IDE (Eclipse, Cloud9, Visual Studio). * Basic ‘know how’ and source control systems and software build process. * Basic AWS knowledge. |
| Requirements  *(List any hardware or software requirements needed for the student to be able to engage with the course material.)* | * Local IDE, either Eclipse, Visual Studio or Cloud9. * AWS account or qwiklabs/linux academy test account |
| Estimatedcourselength  (Standard non-certification courses are around 2-3 hours in length. Introductory courses can be as short as 1 hour) | ~ 3 ½ - 4 hours |
| Cost  (Outline the anticipated cost for students to complete this course) | The tools required in the course are all open source and don’t cost anything.  Students will need an AWS account to test and try out the examples, this will produce costs depending on how much and how long the AWS services will be used.  CodePipeline for example is free to use for one month, if the students use it any longer, it will cost money.  It is hard to estimate, but I would guess about 50-100 $ AWS costs. |
| Permissionsitems  (If you think your course may need any third-party permissions items i.e. stock photos, then give a brief description here.) |  |
| Versioncontrolsystem | Will the students of your course require access to code snippets, scripts, etc.?  **Yes**  No  *At ACG, we use GitHub as our preferred version control system. All code required for your project needs to be hosted on the ACG GitHub account. If required, at the start of your course production, we will set up a repository for your course, and give you Write access to the repository.*  *Please list your GitHub account name or associated email address:* [Wolfgang.unger@gmx.de\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_](mailto:Wolfgang.unger@gmx.de__________________________)  <https://github.com/wolfgangunger> |

Course specifics

|  |  |
| --- | --- |
| Numberofchapters  *(Please give a rough indication, this can change over time as the course develops)* | 5 |
| Structure  (Please give a high-level breakdown of the course structure.  On the ACG platform, each section equates to a chapter, with many individual lessons under each chapter.  Each lesson (individual video) should ideally be around 5 to 10 minutes, and no more than 20 minutes.  The chapters and lessons listed here are for illustrative purposes only; adjust the number of chapters and lessons as required.  Note: some lessons may be theory only, and others will have a demo portion. For lessons that don’t have a demo, this section does not need to be added.) | Chapter 1 Hello  Ch01\_L01 – Hello Cloud Gurus  Face2Face Hello Cloud Gurus Video  Demo Description: No demo  Ch01\_L02 – Introduction, course overview  Lecture Description: Introduction and Course overview  Demo Description: No demo  Chapter 2 CodeCommit  Ch02\_L01 – Introduction Chapter Overview  Lecture Description: Overview of the chapter  Demo Description: No demo  Ch02\_L02 – create first CodeCommit Repo  Lecture Description: Create first Repo, push and pull on this Repo  Demo Description: create a repo, pull and push.  Ch02\_L03 – browse files in Repo  Lecture Description: Understand how to browse the files in the repo and also edit them in the web console  Demo Description: demo on the topic.  Ch02\_L04 – Integration with VisualStudio and Eclipse  Lecture Description: How to integrate CodeCommit in VisualStudio and Eclipse  Demo Description: How to integrate CodeCommit in VisualStudio and Eclipse  Ch02\_L05 –Triggers and webhooks  Description: How to integrate/create webhooks and triggers for CodeCommit  Demo Description: Hands on example on triggers and webhooks  Ch02\_L06 – Branches  Description: How to create, update and merge Branchs in CodeCommit.  Demo Description: Hands on example on Branches.  Ch02\_L07 – Pull Requests  Description: How to create, review and release Pull Requests in CodeCommit.  Demo Description: Hands on example on Pull Requests.  Ch02\_L08 –Summary  Description: Summary on the chapter  Demo Description: No demo  **--- optional – depending on course lenghth – probably not suitable because of the course lenght**  Ch02\_L09 – User Preferences  Description: Information on User Preferences for CodeCommit.  Demo Description: No demo  Ch02\_L10 –Troubleshooting  Description: How to troubleshoot CodeCommit  Demo Description: No demo  Chapter 3 CodeBuild  Ch03\_L01 – Introduction, Overview  Lecture Description: Introduction to CodeBuild and chapter overview  Demo Description: No demo  Ch03\_L02 – Providers  Lecture Description: Lecture about source providers (codeCommit, Github..) in CodeBuild  Demo Description: no demo  Ch03\_L03 – First CodeBuild Project  Lecture Description: First hands on example on a CodeBuild Project, Java App example  Demo Description: Hands on example on CodeBuild Project (Java)  Ch03\_L04 – Docker example project  Lecture Description: Example on how to build a docker image with CodeBuild  Demo Description: demo on docker build project  Ch03\_L05 – Input sources & output artifacts  Lecture Description: Understand (multiple ) input sources and (multiple ) output artifacts in CodeBuild  Demo Description: no demo  Ch03\_L06 – Static website example project  Lecture Description: Example on how to build a static website in S3 with CodeBuild  Demo Description: demo on build project  Ch03\_L07 – Lambda example project  Lecture Description: Example on how to build a Lambda function with CodeBuild  Demo Description: demo on Lambda build project  Ch03\_L08 – Monitoring CodeBuild  Lecture Description: How to monitor CodeBuild with CloudWatch  Demo Description: demo on monitoring in cloud watch  Ch03\_L09 – Env Variables  Lecture Description: Understand env variables in CodeBuild  Demo Description: no demo  Ch03\_L10 – Summary  Lecture Description: Summary on the chapter  Demo Description: no demo  **--- optional – depending on course length**  Ch03\_L11 – Troubleshooting  Lecture Description: How to troubleshoot CodeBuild projects  Demo Description: no demo  Ch03\_L12 – Local test with CodeBuild Agent  Lecture Description: Run and debug projects local with code build agent  Demo Description: demo on codebuild agent  **--- optional – depending on course lenghth – probably not suitable because of the course lenght**  Ch03\_L13 – Command Line Reference  Lecture Description: CLI Reference for CodeBuild  Demo Description: no demo  Chapter 4 CodePipeline & CodeDeploy  Ch04\_L01 – Introduction, Overview  Lecture Description: Introduction and Overview on CodePipeline & Code Deploy and the chapter  Demo Description: no demo  Ch04\_L02 – First pipeline with CodeCommit Repo  Lecture Description: Create the first CodePipeline with CodeCommit as source, building a Java App  Demo Description: demo on pipeline creation  Ch04\_L03 – Create ECS Deploy pipeline  Lecture Description: Create a CodePipeline to deploy a app to ECS  Demo Description: demo on pipeline  Ch04\_L04 – create a pipeline with CloudFormation  Lecture Description: Create a CodePipeline with CloudFormation  Demo Description: demo on pipeline creation  Ch04\_L05 – Create android app pipeline  Lecture Description: Create a CodePipeline to build and deploy a android app  Demo Description: demo on pipeline  Ch04\_L06 – Stages (maybe earlier)  Lecture Description: Lecture about Pipeline stages  Demo Description: demo on how to create different stages  Ch04\_L07 – CloudWatch events & webhooks as change triggers  Lecture Description: How to use cloud watch events and webhooks as change triggers for CodePipeline  Demo Description: demo on how to use change triggers  Ch04\_L08 – Summary  Lecture Description: Summary on the sections  Demo Description: no demo  **--optional – depending on course length**  **--- optional – depending on course lenghth – probably not suitable because of the the course lenght**  Ch04\_L9 – Command Line reference  Lecture Description: Explaining the command line  Demo Description: no demo  Ch04\_L10 – Create Alexa skill pipeline  Lecture Description: Create a CodePipeline for a Alexa skill  Demo Description: demo on pipeline  Chapter 5 Summary, wrap up and next steps  Ch05\_L01 – And that’s a wrap!  Lecture Description: Summary of what has been covered throughout the course and wrap up of content. Brief description of next steps from here. |

|  |  |  |  |
| --- | --- | --- | --- |
| Course filters  (Please select the correct filters that will apply to the course. These are the product filters used on the A Cloud Guru platform to categorise your course.  If you do not feel that your course can be categorised under these filters, please list them under ‘Other’). | Vendor | | |
| **AWS**  **Docker**  Other  (please list): DevOps\_\_\_\_\_\_\_\_\_\_\_\_ | Microsoft Azure  Linux | GCP  Salesforce |
| Primary topic *(please select only one)* | | |
| API Management  **Compute**  Data and Storage  Digital Transformation  High availability  Identity  Messaging  **Operations**  Serverless  Other  (please list): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Artificial Intelligence  Configuration management  Cost and Billing  Disaster Recovery  High Performance Organizations  Internet of Things  **Monitoring**  **Programming**  Voice | Certification  **Containers**  **DevOps**  Governance, Risk and Compliance  Machine Learning  Networking  Security  Web |
| Secondary topics*(select as many as required to represent your course)* | | |
| API Management  Compute  Data and Storage  Digital Transformation  High availability  Identity  Messaging  **Operations**  Serverless  Other  (please list): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Artificial Intelligence  Configuration management  Cost and Billing  Disaster Recovery  High Performance Organizations  Internet of Things  Monitoring  **Programming**  Voice | Certification  **Containers**  **DevOps**  Governance, Risk and Compliance  Machine Learning  Networking  Security  Web |
| Roles | | |
| **Developer**  **Executive**  **Product Manager**  Other  (please list): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | **Solutions Architect**  **SysOps Admin**  **Program Manager** | **DevOps**  Non-technical  **Team Lead** |