Welcome!!

Game Development - Advanced Programming 2019/2020

INFO-6016 - Network Programming
INFO-6025 - Configuration & Deployment

Who Am I?

Lukas Gustafson

What do I do?

Programmer

Tech Lead

mikutech

INFO-6025

Configuration and Deployment

Lukas Gustafson

Today's Lesson

01 Course Information Sheet

02 Expectations

03 Course Goals

Course Description

This course will examine game setup, testing, customization, patching and getting the game to the publisher/player.

Topics will include software delivery, patches and updates, scripting, user configuration, persistence (saving and loading), debugging interfaces and integration with existing deployment and publishing technologies and methods.

Hands-on, practical experience will also be provided in selected topic areas.

Projects & Exams

Tentative projects and exams timeline

Project 1
White Box &
Black Box
Test Cases

September

Project 3
Localisation with
VS & XML Data
Loading

October

Project 4
Installer with
Nullsoft NSIS
Installer

November

Final Exam

December

September

Project 2
XML Loader
and Writer

October

Midterm Exam

MS/VC Build Project

December

Project 5
Registration Keys
License Keys
Registry Entries

Method of Evaluation

Grading for this course will be determined as follows:

60%

Projects
There will be 5 projects. This means each project is worth 12% of your final grade.

40%

Exams
The Midterm Exam and Final Exam are each worth 20% of your final grade.

Submitting your projects

Read Me

All projects **MUST** include a **ReadMe** file. This file must include instructions that explains:

- How to Build your project.
- How to Run your project.

If a project is submitted without a completed ReadMe file it will not be accepted until a proper ReadMe file is submitted.

Video

When submitting your project, include a video of you demoing your project. In this video show and explain how you earned the marks.

You can also use this video as part of your portfolio!

OBS Studio is a free tool for capturing audio and video.

Late Projects

-15% Per Day

What is this course going to cover?

Deployment Process / Publisher Requirements

Test Cases / White & Black Box Testing

XML / Loading XML Data

MS Build Process

Batch scripts / MS Build Scripts

Installation UI-Art Integration

What is this course going to cover?

NSIS / Nullsoft Installer

NSIS Scripting

Registration Keys / LIcense Keys

Deployment / Gold Master Submission

Web Deployment

Core Activities

- Requirements
- Design
- Construction
- Testing
- Debugging
- Deployment
- Maintenance

Methodologies

Waterfall

Agile

Etc.

Methodologies

Selecting the methodology is based on the company and/or project needs. Each framework has its strengths and weaknesses.

Software Deployment

The general deployment process consists of several interrelated activities with possible transitions between them.

Deployment Activities

Release

The release activity follows from the completed development process. It includes all the operations to prepare a system for assembly and transfer to the customer. Therefore, it must determine the resources required to operate at the customer side and collect information for carrying out subsequent activities of deployment process

Install and Activate

It should make all the supporting systems ready to use (Not to be confused with the common use of the term activation concerning a software license, which is a function of Digital Rights Management systems).

Deactivate

Is often required to perform other deployment activities. I.e. a software system may need to be deactivated before an update can be performed.

More Deployment Activities

Update

Replace an earlier version of all or part of a software system with a newer release

Built-In

Mechanisms for installing updates are built into some software systems. I.e. anti virus systems

Version Tracking

Help the user find and install updates to software systems installed on PCs and local networks

Uninstall

Is the inverse of installation.

Publisher Requirements

PlayStation

http://www.playstation.com/en-us/develop

Xbox

http://www.xbox.com/en-ca/developers

INFO-6016

Network Programming

Lukas Gustafson

Today's Lesson

01

Course Information

Syllabus

02

Expectations

Code, running your project and deadlines

03

Lecture

Command Line and Source Control

04

Workshop

Set up repository

Expectations

Documentation

Functions should be documented.

Variables should be documented.

Code Style

A consistent coding style should be used.

Submitting your projects

Read Me

All projects **MUST** include a **ReadMe** file. This file must include instructions that explains:

- How to Build your project.
- How to Run your project.

If a project is submitted without a completed ReadMe file it will not be accepted until a proper ReadMe file is submitted.

Video

When submitting your project, include a video of you demoing your project. In this video show and explain how you earned the marks.

You can also use this video as part of your portfolio!

OBS Studio is a free tool for capturing audio and video.

Late Projects

-15% Per Day

What is this course going to cover?

BSD Sockets (TCP and UDP)

Serialization, Deserialization and Message Framing

Google Protocol Buffers

Introduction to 4 different databases

Service Oriented Architecture

What is this course going to cover?

Asynchronous Programming (Promises)

Connection (TCP) vs Connectionless (REST)

Scaling (Amazon EC2, Sharding, Load Balancing)

Client-Server Game Architecture

Brief Introduction to Command Line

Cmder

cmd.exe | node.exe | cmd.exe | cmd.exe

```
C:\Users\sakul\fanshawe
λ ls
readme.txt  tutorials/

C:\Users\sakul\fanshawe
λ touch newFile.txt

C:\Users\sakul\fanshawe
λ ls
newFile.txt  readme.txt  tutorials/

C:\Users\sakul\fanshawe
λ pwd
C:\Users\sakul\fanshawe

C:\Users\sakul\fanshawe
λ cd tutorials

C:\Users\sakul\fanshawe\tutorials
λ ls

C:\Users\sakul\fanshawe\tutorials
λ cd ../

C:\Users\sakul\fanshawe
λ mv newFile.txt tutorials/newFile.txt

C:\Users\sakul\fanshawe
λ ls
readme.txt  tutorials/

C:\Users\sakul\fanshawe
λ
```

Brief Introduction to Command Line

Shell Examples

- bash
- csh
- zsh
- eshell

Typical Format

```
$ Command arg1 arg2 -option1 -option2
```

ls

ls	List files in the current working directory
ls -l	Lists files in a list with extra info

cd

cd	Change directory
cd -	Go to last directory

mkdir

mkdir	Creates a directory
mkdir -p	Creates a nested directory

rm

rm	Removes a file
rm -r	Removes a directory, and all of its contents

mv

mv Moves a file to a new location

cp

cp	Copies a file
cp -R	Recursively copies a directory

Introduction to Git

Git

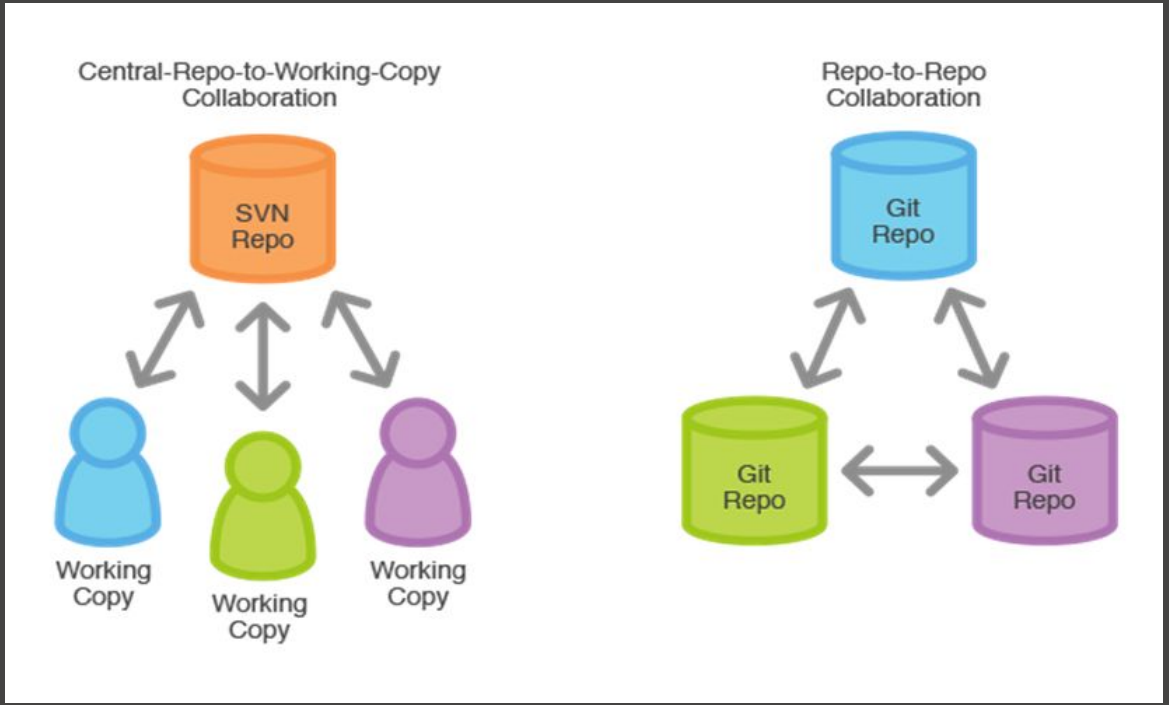
Git is a distributed version control system that manages the history of your project.

All projects **MUST** be done in git.

We will look at only basic theory, and then get straight to the pragmatic benefits of git (Setting up a project, committing, branching, pushing, cherry-picking, reverting.)

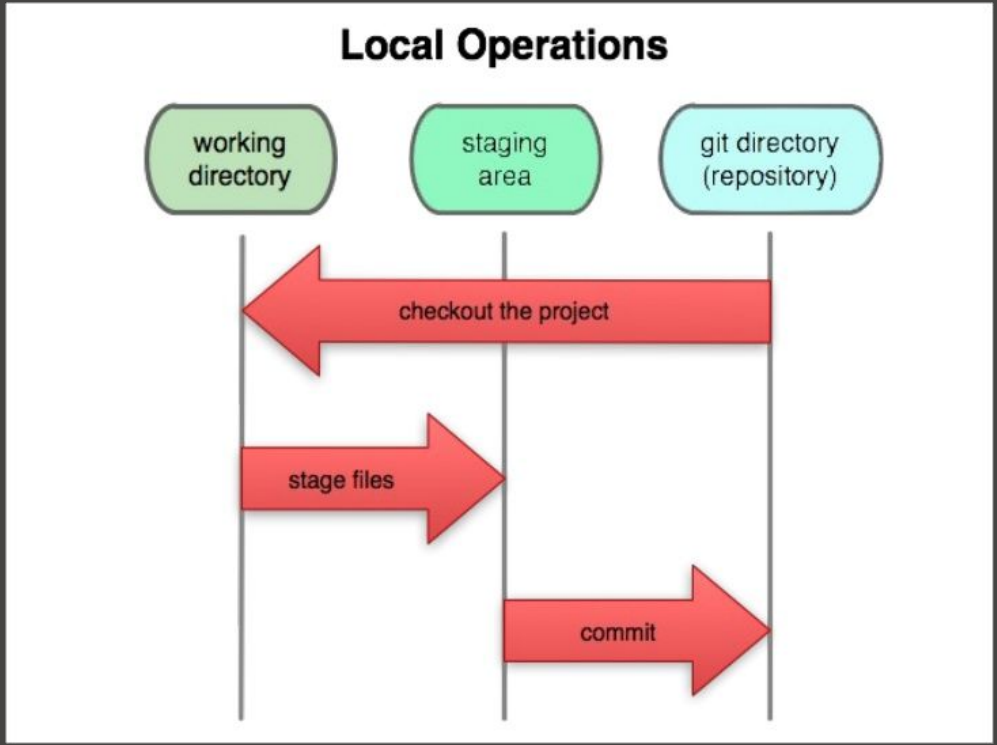
Basic Git Theory

Subversion vs Git



Basic Git Theory

Local Operations



Things We're Covering Today

01 **Staging**
Add your local changes, new files, or deleted files to the stage.

02 **Committing**
Commit your changes to your local repository.

03 **Branching**
Create new branches, and change to another branch.

04 **Pushing to origin/upstream**
Publish your changes to the remote repository.

05 **Tagging**
Tag commits with labels.

In Class Workshop

The easiest way to learn git is to practice, let’s install git and start!

Install Git

http://www.git-scm.com

Create Github Account

https://github.com/

Create a Repository

Configuring Git

git config --global user.name “Git Rules”

git config --global user.email gitrules@git.com

~/.gitconfig

C:\Users\$username$\.gitconfig

Key Generation for SSH

ssh-keygen -t rsa -C “my-email@gmail.com”

Initializing a Project

git init .	Initializes a repository in the current directory
git init project_name	Initializes a repository in a new directory

Cloning

We can clone external repositories via:

-http
-ssh
-local directory

git clone URL/PATH

Staging Files

git status	see unstaged files
git add filename	stage file
git reset HEAD file	unstage file

Committing

git commit -a Add all tracked files.

git commit -m “some small message”

git commit

git commit --amend

Branching

git branch	List branches
git branch “branch_name”	Branch off to new branch
git branch -d “branch_name”	Delete branch locally

Changing Branches

git checkout “other_branch_name”

git checkout -b “new_branch”

git checkout sha-1 Checks out a specific commit

Merging

git merge “other_branch_name”

Adding Remote repositories

git remote add “remote_name” URL/PATH

git pull “remote_name” “branch_name”

git remote remove “remote_name”

Pushing/Deleting Remote Branches

git push origin “branch_name”	push changes to remote branch
git push origin :”branch_name”	Deletes a remote branch
git push origin --delete “branch_name”	Delete a remote branch

Tagging

Pick a commit and add a name to it.

git tag	Lists tags.
git tag “tag_name” sha-1	Tags a specific commit

Git Short Review

git add

git commit -a

git push

git log

git checkout

Git Short Review

git add	Add files to stage.
git commit -a	Push changes to repo.
git push	Push the repo to a remote location.
git log	Show a list of changes.
git checkout	Change branches.

In Class Workshop

Partner up with the person sitting next to you.

01 **Create Repository**
Log on to Github. Create a new repository. This can be public or private.

02 **Invite**
Send an invite to your repository to your partner.

03 **Clone**
Both of you clone the repository to your own computers.

04 **Initial Commit**
Get the first member to add a file to the repository. Commit the change, and push to the remote repository.

05 **Modify**
Get the second member to change the file. Commit the change, and push to the remote repository.

06 **Synchronize**
Both of you pull from the remote repository to synchronize all of the changes.

Questions?

Ask questions; don’t make assumptions. - Angela Ahrendts

Resources

OBS Studio: https://obsproject.com/

Cmder: https://cmder.net/

Google C++ Style Guide: https://google.github.io/styleguide/cppguide.html

Git: https://git-scm.com/downloads

Github: https://github.com/

Credits

This template was created by GoogleSlidesppt.com

Inserted pictures in this template were created by pixabay.com

References

Wikipedia: https://www.wikipedia.org/