

Meeting Logs

Format:

Date, duration

Topic, people present

Notes

12. Jan 2023, 12:00-13:30

Project introduction, Richard, Robin, Viktor

discussed project methods and tools

decided on scrum, jira, bitbucket, office365

discussed timetable, project scope, brainstormed game mechanic ideas, contracts

16. Jan 2023, 11:40-12:30

Project planning, Richard, Robin, Viktor

Planning with Richard, tried to set up jira

16. Jan 2023, 13:00-13:45

1st meeting with advisor, Mariusz, Robin, Viktor

try to isolate our work from richard's

things richard does is out of scope

don't need commits as proof

we have to write about the game, so hard to keep secrets

keeping bachelor secret hinders our ability to show off to employers

can send nda to advisor

nda should not be too restrictive

to get A:

not just impliment, talk about innovation

compare, read papers, review other games/solutions

mayb one central design problem

maybe focus on a central "theme"

ie optimization, game economy, balance, game theory, negotiations

don't spread it too thin

"Innovative Approaches for More Affordable Computer Game Development"
too researchy?

demo subsystems, screenshots, not runnable code
normally need to include code and repo, not public

report, include stats, lines of code, how big is the project, technical details

23. Jan 2023, 17:00-18:00

1st sprint meeting, Richard, Robin, Viktor

generate terrain:

not too much altitude variation with adjacent tiles

RGBA, A is blank

ranges of color represents terrain types and heights, (adjacent tiles? probably not)

need to be able to get altitude based on color

altitude doesn't need to be smooth

meeting again on friday

discuss:

sprint, terrain generation and project plan

backlog, 5 points

nda

new laptop? graphics card is important, i7 gen or ryzen equivalent

30. Jan 2023, 11:30-12:00

Project plan document, Mariusz, Robin, Viktor

iterate plan document, latest version will be appendix

update, changes should be logged

clarify AI terms with richard

how frequent are meetings, contingencies

hopefully weekly sprint meetings, organise discord channels

gant diagram is important, make backlog with richard, priorities

nda: only send to mariusz if we have questions

risk: add employer becomes unavailable, critical delays

how to resolve conflicts in the group

no sources needed for info about the company, no sources in the plan

1.3 is too technical, should be about what we should not do ie. unity not unreal

2.2 is about high level stuff ie. not research, focus on procedural generation

weekly meetings on 13:00 monday

30. Jan 2023, 17:05-18:00

Map generation specifics, Richard, Robin, Viktor

maps must generate one tile (chunk) at a time

why? more control over generation, generating based on game state, being able to continue a game with updates

should tile know its surroundings? Probably

config map is the most important thing

jira backlog

nda

sprint meetings mondays 9:15

9. Feb, 21:00-21:45

Pipeline specifications, Richard, Robin, Viktor

no meetings on monday mornings

algorithms should work for both team and runtime? regenerative?

rgb map should be the entire world, no need to be procedural

rgb map is only for biome borders, water, and altitude

next steps: tweak biome look, maybe use diamond square, rivers and lakes

rgb map is used by unity, after this step we don't generate the whole map at once

lazy flood fill for details like trees and stuff, place resources

13. Feb, 21:00-21:50

Future meetings, Richard, Robin, Viktor

Discussed how future meetings would be sparse due to scheduling issues

Showed our progress so far

23. Feb, 14:30-15:20

Project state, Mariuz, Robin, Viktor

- Communication issues

- Sparse meetings

- Employer forgets meetings and doesn't show up.

- Miss-communication

+ Unclear on if it's generate pr game on the go, or used to make a set of pre-made maps.

- > Says we should make assumptions while we wait and inform employer.
- > Pre-made vs. on the go is very different.
- Unity
- Planning to port to unity.
- Unsure if we should integrate with specific unity program or general unity.
- > Says we need to get clarification on whether generation happens on same machine as unity integration or not.
- > Asked about our unity experience.
- Both used it in game programming last semester.
- Sprint planning
- We have no proper issue board, just picking ideas and rolling with it as we go.
- Too little information to make clear lists. No idea what we will do, come up with ideas as we go.
- > Risk that we don't have a working product.
- > Need some plan of what we need to arrive at and general idea on how to get there.
- > Don't need to plan each point, just 2 of you, but need enough to get the required features before we're done.
- > Prepare a suggestion on final req-spec, suggest it to employer, fill the blanks with assumptions.
- > Process is very iterative, research -> idea -> results -> repeat, but still need to fill the requirements.
- > Can generalize and translate to his desired format as an interface, but need to understand what he wants so we can adapt properly, otherwise risk not having the data needed to fit his format.
- > Figure out if you're making a library, or custom made for the game.
- > Need to know what the final product will be so we can prepare how to demonstrate.
- Plan is a generalized library that fits employer's need and an interface to convert into his desired format.
- Demonstration planned to be some generated maps in the format of the game.
- > Should measure the different algorithms in different ways.
- > Ex.: Interesting, realistic.
- > Maybe some objective metrics.
- Performance, scalability

- > Could generate terrain and render it, then ask human subjects whether they think it was procedural or designed by a human.
- > Rendering the terrain in 3D might help us see problems or get ideas.
- > Consider consulting people with experience in terrain/biology related knowledge for pointers that normal people can't see.
- > Geographers.
- > Maybe literature.
- > Different priorities for research vs game.

27. Feb, 17:00-18:00

Unity integration, Richard, Robin, Viktor

Planned to integrate into his unity code

Richard wants to work together tomorrow

(Retroactive note: Richard did not have time to work with us after all)

2. Mar, 14:00-14:40

Project state, Mariuz, Robin, Viktor

- Progress going well.
- Got meeting with Richard.
- Got a lot of new info.
- Showing Mariuz the unity program.
- Showing the new terrain based representation.
- > Asks how the texture maps to terrain
- 1 height map
- 1 texture based on tile colors
- Changed it so rivers cut the terrain so it doesn't go above land.
- Smoothing terrain
- > Still some ruggedness on the texture.
- Explaining that the visualization is meant to be client-side
- Clients can use textures and alter how it works in the game.

- We moved all the code into unity.
- Might get to work with Richard to integrate the program into his game.
- Project agreement
- Not gotten a meeting to go over it with Richard yet.
- Only Norwegian version on BB.
- Project rapport
- Should we start slowly with it already?
- > Yes. More fun to play with code, but rapport is main source for grade.
- Documenting throughout gives better info than figuring it out later on.
- Don't need to make full document yet, just start adding documentation of progress, screenshots, decisions as we go, format it later on.
- The stuff we have in the #progress channel works good as documentation.
- > Distinguishing who made what
- We should isolate what we made from what Richard made or we made with him to make it clear.
- For when we can't isolate it, ex.: we makes changes to his code, we should make a diff file to show what we did.
- If he changes our code, hard to show exactly what was who, but we should try to document what he did.
- Ideally provide a repo so they can check the commit history if they're unsure.
- Probably won't be checked too much, but best to be safe.
- > Compare our project to similar projects that did procedural generation.
- Find others who did things similar to what we did, compare and discuss.
- Is a positive to do this research and present it, not a negative to not invent everything ourselves.
- > Asks what we're going to do next.
- Temperature generation by simulating sunlight and shadows.
- Might work on a wind-layer.
- Simulate wind on map.
- Possibly use it to transport humidity.
- Make map vary less or more in altitude in different areas instead of having everything have the same spread.

- Store and pass the surface area of lakes so we can make it look similar to oceans.
- Storing biomes instead of just biome types.
- > Asks if we're planning to work with changes to map over time or just generate a finished map.
- Possibly have wind and time of day affect layout.
- More of a later on thing, need to make good maps first.
- > When we talk with Richard, we should get a prioritized list of features so we know what to work on.
- > On rapport: Not writing it full time, just adding stuff from time to time while working on project, full-time ca. 5 weeks before dead-line. (2-3 weeks for writing, 2 for reviewing)
- > Project agreement: thinks deadlines are flexible, but better safe than sorry, so check with Tom.
- > NDA: only important for Richard, so doesn't matter for NTNU if we get it late or never get one.
- > Measure difference in performance and ram use between objects pr. tile and a single terrain object.
- > Not that important, but would look nice to have as data on the rapport.
- > Should figure out how unity terrain works and describe it in the rapport.
- > Ex.: Smoothing the peaks on the map.
- Shows Mariuz the altitude smoothing.
- Mariuz likes that we have both smoothing rendering and smoothing the actual tiles.
- > Is curious about how the game using these tiles works, wants us to ask more about it from Richard next time we have a meeting with him.
- Showing Mariuz the diagram of the map pipeline.
- Showing how it can be used flexibly and both for on the go map generation and custom made maps.
- > Might want to choose what we provide of the box, ex.: rescaling the map, preview rendering.
- Will provide flexible base class for converting between common and unity format.

10. Mar, 14:10-15:00

Project state, Mariuz, Robin, Viktor

- Still no meeting w/ Richard
- Didn't hear from him until today, wants to meet in the weekend, but went dark before a date.
- Asked Tom about project agreement.
- No English template.
- Richard has signed before.

= Richard will just need to sign the Norwegian.

- Current work

- Making the GeneratorBuilder customizable in unity inspector.

- We want dropdowns for each subclass of step generators.

- Focused on tools for tweaking the map at runtime.

- Showed the inspector tool to Mariuz

- Showed how it works, how it's made with reflection.

- Showed how the inspector struggles to update, needs to swap out and back in.

- Didn't get to show a map being generated due to unrelated errors.

- Showed Mariuz the wind system.

- Showing how the wind reacts to altitude.

- Funnel effect.

- Wind doesn't do anything at the moment, but thinking of using it for things like heat and humidity.

- Thinking of changing it from left to right to be more dynamic.

= Convection wind

= Heat moves air up causing wind. Moves up the valley.

= Can use wind for stuff like erosion.

- Running out of things to do without input.

- Unsure if the tool we're making will be useful for Richard or if he has his own system for making and playing with maps.

= We might need to make our own choices and start looking at alternative solutions to compare for our rapport if we don't get enough to do from Richard.

= Might need to make our own self made demo to let people play with the programs we've made.

= Might need to make our own choices instead of waiting for Richard.

= Give him deadlines, then do our own thing if he doesn't respond so we have stuff to do.

= Looks good

= Still have some potential for adding interaction and optimization

= Ex.: Progress bar as map is being generated.

= Maybe add unit tests to the program to detect exceptions.

- = Asking if we're doing things with light.
- We have very basic sun simulation, affects temperature.
- Tweaked lakes
- Old is mountain -> {lake, ocean}, ocean -> glacier
- New is that lakes make rivers down and don't need a mountain.
- = Dilemma of sharp or smooth edges.
- = Different rock/soil types, sharper/smoothier transitions between altitudes or biomes.
- = In the rock river, caves into the ground and comes out elsewhere. (Rare, but happens reasonably often)
- = Water temperature
- = Static map, we generate a map, then it stays that way, no change as it runs.
- = Should ask Richard if it should stay the same or change over time, ex.: Erosion, seasons, flood.
- Maybe player made changes, ex.: Dams, redirect rivers
- = Plan for next week
- See if we can meet with Richard.
- Wind and sun maps.
- Maybe basic geology.
- Iterate.
- Canyons & Ravines, erosion
- Nature and resources

11. Mar, 16:50-17:50

Project agreement, Richard, Robin, Viktor

Filled in project agreement, richard doesn't want shared ownership

Also showed more of our progress

13. Mar, 15:05-15:35

Project agreement, Mariuz, Robin, Viktor

- Going over Richard's answers

- Funding: Bitbucket.
- Chose that employer retains full ownership of results.
- Reasoning: Investor related things.
- = Mariuz disagrees:
- = Thinks we should have co-ownership with Richard gets full usage rights.
- = Otherwise we're left with nothing.
- = Give him non-transferable right to freely use the results so he has no risk.
- How should we sign the document?
- = No digital solution in NTNU as far as he knows.
- = Should just physically sign it and deliver it as a pdf to BB.
- = Document never used unless there is a legal dispute.
- = NTNU doesn't normally own our work.
- = Only us and him to argue over licensing rights.
- = University has no rights over our results even if we pick the option where we retain rights.
- = We have full right to not publish part of our code in the thesis, relevant for if we make code that integrates our code with his and it contains proprietary info.
- We filled out the remaining fields info for things like contact teacher and the IDI department.
- "Name of the project"-field, do we need to fill that out right now?
- = 3 options: Original name, current working name, or name when we finish the project.
- = Go with current working name.
- Signature of contact teacher and head of department.
- = Upload without their signature, they sign only if there is a legal dispute.

16. Mar, 14:00-14:45

Project state, Mariuz, Robin, Viktor

- We sent Richard a message about the ownership decision.
- No response yet.
- Hope it won't be a problem that the agreement is taking time.
- = Not good that it's late, but not gonna be big problems for us since we're doing the work to get it done, just waiting for Richard

- Question about agile.
- We wrote that we'd do agile, but so far we've worked pretty much barbarian.
- Little plans, basic to-do list.
- Is it a problem if we have don't have planed sprints, but we have what we did each sprint.
- = Yes and no, Agile is meant to adapt, we have some form of plan, try to use a different name other than barbarian.
- = Need some sort of plan for 2 reasons.
- = 1) So we can change the process as we go to improve it.
- = 2) Predictability. Don't know when we're gonna be done with what in agile, but have some degree of predictability. Able to estimate how long something will take based on estimated difficulty and past productivity.
- = Not following agile is okay, but need to describe the process we're using and how it changed over time. Also need to describe how we measured productivity/estimated task costs.
- So maybe we should make some kind of plan for what kind of things we plan to focus on next sprint?
- = Yes, don't need to be full agile, but good idea to have some documentation, also document differences between planned and actual progress.
- What we've been working on.
- Viktor: Mostly the windmap.
- Left to right, affected by direction and altitudes.
- Tweaked a lot of things to improve it.
- Measure direction.
- Water surface counts as altitude for the windmap.
- Messing with constants and formulas.
- Wind becomes very thin streams, want more natural spreads.
- = On the narrow wind streaks, depends what it's used for. If we use it for a particle system, the narrow streams might be pretty good.
- = Should discuss with Richard how he wants to use the wind system to decide how it should work.
- So far, plans to use it to spread humidity and temperature.
- Low wind => less humidity spread => more extreme temperatures.
- = For humidity spread, might not want the thin streams of wind.
- = Should have some diffusion system.

- Robin has been working on new system for encoding the data.
- Can decide what data to encode.
- Can change how many bits to use for each.
- = Gets what we're doing, but would add another layer of indirection.
- = The data we store has nothing to do with imaging, we just use it to store data in a grid format.
- = Should just pass the data directly into a binary file, less wasted space, less splitting into images.
- = Recommends not going below 32-bit chunks, processing costs are higher than the storage rewards.
- = We could use bitwise if we don't want random access, but it's a limitation.
- Suggesting to do one parser for padded and one without.
- = Likes the idea, could do comparisons between the 2 formats.
- Talked to Richard about doing sub-biomes
- Potentially use factors to divide biomes into sub-biomes.
- = Should give parameters for client to control what rules decide sub-biomes. Configure how much different factors control what sub-biomes to pick.
- = Should compare with other designs.
- Maybe we should do more research for a bit, struggling to choose what to do, little info from Richard.
- = We're going to need to do more research, do reading, comparisons, find out a topic for the thesis.
- = Usertests, experiments, testing on people.
- = Important chapter and element of the thesis, part that Richard won't help us with.
- = Try to find the research area we want to explore.
- = Thinks how to generate interesting biomes and terrains is a good idea.
- = Which terrains are realistic or not, plausible, interesting, practical.
- = Geology
- = Had a group that re-implemented and refined their entity system and compared them and research how other people made it.

30. Mar, 14:00-15:25

Project state, Mariuz, Robin, Viktor

- Project agreement
 - Some responses from Richard, want to know if it matches the agreement.
 - Richard wants full ownership, give us free usage right, including open sharing and stating involvement.
 - Isn't that what we'd have anyways with our proposed agreement?
 - = Yes. He might misunderstand what shared ownership means, thinking he needs us to agree to any modifications he might make to the original code.
 - = Co-ownership is when all parties must agree to any usage of the item, shared ownership is all parties can do whatever they want to it.
 - = What you can do is explicitly outline in the agreement that he can do whatever he wants with it and we can publish it and state our involvement.
-
- The amount of time we need to spend.
 - Said somewhere that we needed to spend a specific amount of time on the project.
 - Is it a hard cap or a guiding cap?
 - = Tricky question.
 - = On one side, not a big deal if we're a little below the limit, some students are more effective than others.
 - = Should be aiming at around 600 hours per person. Probably fine with 500 hours per person.
 - = Shouldn't make up numbers, but should be clear what took more or less time than expected.
 - = Treat it as a guideline, but if we're under it, elaborate what took less time than expected.
 - = If you just give hours without elaborating what we did and what took more/less time than expected, they treat the numbers as how hard we worked.
 - = ECTS points, each point is worth 28 hours
 - = Don't stress it too much, most important is how much you work.
-
- Richard sent us some confusing messages.
 - Told us that the bachelor was to make a solution for him, not to make a generalized tool.
 - We've been telling him for a while now that we're making a generalized tool and he hasn't said anything against it until now.
 - Told us to put more effort into integrating the code, but still hasn't given us code to integrate with.

- Still unsure if he wants a map maker, or an automated map generator, or even a pr chunk map generator.

= You need to clarify with him.

- Also asked him about whether we had to store it as an image or could store as binary file.

- Unsure whether he's confused since the maps won't be human readable anyways, but we have visualizers as a separate thing.

= Think he misunderstood what we meant, that he would be fine with binary file, with visualizers separately, but thinks we mean skipping map visualization entirely.

= Should clarify it with him.

- Show some of the progress we've done.

- Some thermal erosion to augment the altitude.

- Compared it to an old smoothened.

- Might work on erosion based on factors like humidity and temperature.

- More ways to tweak/adjust altitudes or other factors.

= Is it possible for us to have a layer that that smoothen altitude to a curve, like exponential.

- yes.

- Explaining how the StepGenerator system works.

- Explaining the new MapAspects management we added.

= Asked about how long these steps take.

- Different generators take different amount of time, for a 1025x1025 map, Perlin or Diamond Square generators for factors usually take a few seconds at most, environmental factor generator usually takes 1-3 minutes depending on the parameters due to it doing more complex simulations.

- Wind generator

- Finally got an initial version, also added effects on temperature and humidity.

- Going to take a break from wind generation for a while, work on other things, maybe come back.

= Should clarify that we're making a generalized toolkit and we can integrate it with his code if we get more information on what we're integrating with.

= Thinks we've done good work and have a good bit to write about.

= Would be good to get him to review the features and make a list of what we came up with together and what we came up with ourselves ideawise.

= Important for reviewers to show what we came up with ourselves and what we implemented that came from him.

- Could make a bunch more step generators and compare if we need more to write about in the rapport.

= Would help add more research to add more different generators and compare their performance and their results.

= For a bachelor, the project is very important, the experiments aren't as important. For masters, it's usually the opposite.

= Adding more research could still help improve the rapport and increase our grade.

- Been reading other projects that do procedural generation.

- Many of them mention feature based generation. User describes some features, the code then fills in the blanks.

- Sounds useful for Richard, but think it would take quite a lot of time.

= Don't know the answer. Maybe have a combination where hints can be added between steps.

= Check with Richard, maybe explore the idea if he wants it.

= How are the other people we read the rapports from doing it?

- In one, the user draws a bunch of curves and then the algorithm adds noise and fills in the blanks of those curves.

- They simulate some erosion with water.

- Splines: They take the curves and convert them into some sort of 3D structure.

= Sounds interesting, but we only have 1 months left, so have to manage our time budget.

= Potentially explore some of the ideas and put stuff we can't do in time into a further work section in the rapport.

= Likes the idea of generating a map, then tweaking it, then running it through some smoothening maps and repeating.

= For the last month, we shouldn't experiment too much, but for now we have some time, so we can experiment a bit, but it is too late to rebuild the whole project at this stage.

= Prioritize Richard first, then experimentation.

= Compare things, make images to compare in the rapport.

= Need images to explain the journey for the reader, explain what things mean, how things effect the outcome visually.

12. April, 21:08-22:14

Project Specifications, Richard, Robin, Viktor

- Agreement

= Mariuz said it's fine to draft our own agreement.

= How has it been going

- Slow recently, partially easier, partial needing clarification.

= Subbiomes

= After running over the biome generation, use flood fill to pick sub-biomes.

- Turns out sub-biomes = nature and resources

- Questioning about visual map saving.

- Only store it as bitvalues, not human friendly.

- RGB values mean nothing to people anyways.

- Have ways to visually depict the map, but they can't store data.

- Why not use binary files?

= Part of the rationale is for modding on disk, recognize the map from the thumbnail.

> Conclusion: json, binary file and a human friendly thumbnail.

- Resources

= Possibly overlapping.

= Resources like wildlife cover multiple neighboring tiles.

13. April, 14:00-14:25

Project state, Mariusz, Robin, Viktor

= Did we get a project agreement?

- Not yet, but had a meeting with Richard yesterday and he said he'd make one soon.

- Should we do a formal complaint about lack of meetings and feedback from Richard?

- Our planning and meetings look bad on the face of it, reasoning being lack of communication with Richard.

- Do we need a formal complaint for reference?

= There is a formal process, but it's probably useless, most projects are like this where there is poor communication with employer.

= Recommend not making formal complaints, recommend elaborating the situation in the rapport.

= Explain the situation, but don't use it as an excuse.

= Explain how it works, explain how we were only 2 and the overhead wasn't worth it.

= Explain how you'd do it if you had to redo the project now and what you've learned (Reflection)

- Maybe show the commit history to show that we worked consistently?

= Should show the history to demonstrate that we worked consistently.

= Show that we didn't work chaotically that the goals were set a bit ad hoc.

- On the meeting with Richard

- We finally resolved the storage system question and settled on using binary file for tile data and adding a thumbnail.

= Should put in the rapport that we convinced the employer to choose a different technology.

- He wants us to work on "sub-biomes", but we finally found out he basically means resources.

- We also have to make it so we can generate a new map using an existing map.

- We just got that done today.

- We also want to make a bunch more generators for people to play with.

- = Did we consider how we're going to evaluate what we've done.
 - = Can use concrete numbers like performance, can do user studies, can ask Richard, can use personal satisfaction.
 - = Ex.: Code quality w/ linter, test coverage, commit history to show consistent work.
 - = Should have some unit-tests/end-to-end tests.
 - = Don't need to do all these things to the same depth.
 - = Evaluation is an important chapter, should plan how to do it ahead of time.
 - = Can use quantitative (concrete numbers), qualitative (expert interviews, ex.: Richard), own satisfaction.
-
- = What features are we working on for the last 2 weeks?
 - Data into binary file
 - Code clean-up.
 - Resources.
 - Unit-tests.
 - More generators.
 - = Can work on some stuff after the rapport, before the presentation, like a demo.
-
- Should we have diagrams like use-cases?
 - = Yes, sequence flows, use-cases, etc.

4. May, 14:00-15:30

Report tips, Mariusz, Robin, Viktor

still no project agreement, it doesn't really matter, don't worry too much about it

hard for us to translate agreement to english, don't need to

if no agreement we have default one, this benefits us, we won't fail or anything

when can you read, he will read thoroughly once, can give quick feedback in between
can also request 1 chapter

our 3-week plan, get tons of text written down, filter and format, margin
get thorough review mid week2 or early week 3

report is not about process timeline, more about how it connects logically
ok to have self-contained section that goes through our progress in time
length of report, might go over 80 pages with tons of images
no image limit, just make sure it makes sense to the reader
no more than 50% images, have enough explanations for visuals

40% is research, super important, at least 10-15 pages, how we designed it
what we learned from research, how we integrated it
can use any source, time is not important in research, what matters is how it connects
can "pretend" we read it early, don't mention when you read it

in general, time is not a factor we need to report!

up to us how to organize the thesis, make sure it has logical connection, details are up to us
intro, background/research, implimentation, evaluation, discussion/future work
get inspiration from other thesis
link to alternative format:
<https://github.com/COPCSE-NTNU/thesis-NTNU>

keep timelog (progress channel) self-contained in process section, keep it brief, with stuff like meetings,
organization

explain in detail in other section, but sort logically ie river generation

mention stuff 3 times:

quick mention by week, explain how we implemented iteratively (sorted logically), exactly what we ended up with

design is abstract stuff before implementation ie. diagrams, architecture, ok to get low-level

process is about how not what, how we organised ourselves, how we tested etc

implementation is about code, the code

development process: agile stuff

doesn't matter what we planned, don't need to mention agile plans

say this: started with waterfall (say rigid, not waterfall), went from rigid to agile

start with the end, then say how we got there, what adjustments we made

requirements: split between functional and non-functional

design: skip history

implementation: show code, but not much

unit tests: maybe skip if we have a ton of benchmarking, argue for why we don't need it

performance tests: charts are nice, include table as appendix, make dedicated benchmarking chapter

what makes a good map?: say more objective stuff ie cartoony vs realistic

evaluation: what would we do differently

discussion: more free flow, less formal, thought about procedural generation or what we could have tried?

conclusion: summary, results, go back to abstract

introduction:

background: employer, us, project in general, what is procedural generation

problem area: what richard saw as the problem

avgrensninger vs rammer: worry about it later, write now, format later

oppgavedefinisjon: about results

målgruppe: game developers, students, people who will use the tool

11. May, 15:55-16:40

Project state, Mariusz, Robin, Viktor

- Contract agreement

- Finally got an agreement from Richard.

- Seems like he gets all the rights and we get nothing.

- = Reads like he gets all the rights and we just get to read, develop and promote it.

- Can we open source it?

- = No, we need full right to be able to give others full rights.

- = Should get him to add in the agreement that we can publish it as open source if we want to release it as open source.

- Bachelor presentation

- 6th june, but doesn't say when.

- = Usually starts at 9am and goes till around 4pm, depends on how many presents, usually has a break in between.

- = Presentations are usually 30 min, but length on how much time they have in total.

- = Best to ask Tom about the organizational questions.

- = Grades set based on rapport before presentation. Presentation usually doesn't lower the grade.

- = Grades are usually released the day after the presentation.

- The rapport

- Struggling with the research part. Unsure how to write it.
- Another bachelor didn't have a research section, but had it spread around the rapport.
- = Some reviewers look more into the product, others look at it more like a mini master thesis.
- = Our project fits both a little, both a product, and some research around procedural generation.
- = We shouldn't worry too much about it, just write what we did, the testing and experimentation we did.
- = Research section is sometimes implied more than explicit.
- = Trying something, examining the results, then trying something different is research.
- Sources
- Thinking about looking through the masters, reading through their sources, and using them as sources.
- Not just copy pasting, actually read what they cite, then use that as sources.
- Would that be seen as plagiarism?
- = Not plagiarism if we use similar sources, as long as we use them as sources, and not just copy what the master said.
- The progress section
- Thinking of using it as a chronological list of what we did.
- Problem is that implementation would just be repeating progress for the most part.
- Hard to dumb certain things down without it being meaningless.
- Would it be a problem if implementation is a repetition of progress with some more detail.
- = If it feels like we're repeating ourselves too much, we should probably change it.
- = What do we want to communicate with the progress part?
- = Could scrap the progress part for being a duplicate. Add a short summary of progress week by week?
- For generators, there is a lot of overlap between progress and implementation, but for some things, like how we save, there are important changes.
- = If there are important changes, they should be integrated in implementation.
- = Recommend moving the progress content into the other 2 sections instead.
- = Don't repeat ourselves too much.
- = Worst thing is not to say something. Better to say it in the wrong place, than not at all.
- = Risk having a B level product, but a D level rapport for not mentioning all the things we did.

= If we tested out rivers without bezier curves and found it was bad, we should still describe the experiments in the rapport. Even if the result was useless, the test is still important.

= Anything we don't mention will basically not have happened.

- The diagrams

= Good diagram

- Planning to delay asking for feedback.

= Remember to spend almost a day on the abstract.

= Abstract is the first impression, gives the reviewer a bias about the quality of our group.

- Looks like our access was changed to read only.

- Unsure if it was manually done, or if it was some automated screw up.

- Tweaked our Unity code to allow any map size.

- Had to make bigger terrain, then cut the obsolete parts with holes.

- Also fixed an issue where the altitude didn't perfectly align with the texture on the terrain.

= Should mention all the tests and refactoring we did to improve the quality of the code and product.

= Should add a clause that makes the product automatically open source when we finish if we wanted to make it open source.

= He still retains all his rights to it, so he can use it freely in proprietary products and modify it even if we release it as open source, since he has it under different license than the open source license.

= Rights are different to authorship.

= Back in the days authorship cannot be transferred even if ownership is transferred.

= Might have changed.