

CS 571 - Data Visualization & Exploration

Final Project & Git for Management

Instructor: Hamza Elhamdadi



UMassAmherst

Upcoming Dates

Feb 13 (Today): Announce Your Project

Feb 14 (Tomorrow): ~~Group~~ Activity 1 Due

Feb 14 (Tomorrow): Homework 1 Due

Feb 28: Project Proposal Due

Project Timeline (dates may change)

Announce Your Project	Feb 13
Project Proposal	Feb 28
Project Review	Mar 3 - 14
Project Milestone	Apr 4
Peer Feedback	Apr 16
Project Screencast	May 2
Final Project Submission + Group Feedback	May 12

Project Proposal

Your Project Proposal Will Include:

- 1. Project Metadata**
- 2. Background & Motivation**
- 3. Project Objectives**
- 4. Data**
- 5. Data Processing**
- 6. Visualization Design (Five Design Sheet Methodology)**
- 7. Must-Have Features**
- 8. Optional Features**
- 9. Project Schedule**

Your Project Proposal Will Include:

1. Project Metadata

This section should include:

- The title of your project
- The name of each group member
- The UMass email of each group member
- The SPIRE ID of each group member
- A link to the project repository on GitHub

Your Project Proposal Will Include:

- 1. Project Metadata**
- 2. Background & Motivation**
- 3. Project Objectives**
- 4. Data**
- 5. Data Processing**
- 6. Visualization Design (Five Design Sheet Methodology)**
- 7. Must-Have Features**
- 8. Optional Features**
- 9. Project Schedule**

Your Project Proposal Will Include:

2. Background & Motivation

This section should include:

- Discuss your reasoning and motivation for choosing the project
 - any background or research interests that may have influenced your decision

Your Project Proposal Will Include:

- 1. Project Metadata**
- 2. Background & Motivation**
- 3. Project Objectives**
- 4. Data**
- 5. Data Processing**
- 6. Visualization Design (Five Design Sheet Methodology)**
- 7. Must-Have Features**
- 8. Optional Features**
- 9. Project Schedule**

Your Project Proposal Will Include:

3. Project Objectives

This section should include:

- the primary questions you are trying to answer with your visualization
 - What would you like to learn and accomplish?
 - What are the benefits of learning/accomplishing these things?

Your Project Proposal Will Include:

- 1. Project Metadata**
- 2. Background & Motivation**
- 3. Project Objectives**
- 4. Data**
- 5. Data Processing**
- 6. Visualization Design (Five Design Sheet Methodology)**
- 7. Must-Have Features**
- 8. Optional Features**
- 9. Project Schedule**

Your Project Proposal Will Include:

4. Data

This section should include:

- Information about the source of your data
 - From where and how are you collecting it?
 - Provide the link to your data source(s)
 - unless you can't (e.g., for legal/privacy reasons)

Your Project Proposal Will Include:

- 1. Project Metadata**
- 2. Background & Motivation**
- 3. Project Objectives**
- 4. Data**
- 5. Data Processing**
- 6. Visualization Design (Five Design Sheet Methodology)**
- 7. Must-Have Features**
- 8. Optional Features**
- 9. Project Schedule**

Your Project Proposal Will Include:

5. Data Processing

This section should address the following:

- Is your data ready-to-go?
- What level of cleanup do you need to do?
- What quantities do you expect to derive from your data?
- How will you implement data processing?

Your Project Proposal Will Include:

- 1. Project Metadata**
- 2. Background & Motivation**
- 3. Project Objectives**
- 4. Data**
- 5. Data Processing**
- 6. Visualization Design (Five Design Sheet Methodology)**
- 7. Must-Have Features**
- 8. Optional Features**
- 9. Project Schedule**

Your Project Proposal Will Include:

6. Visualization Design (Five Design Sheet Methodology)

This section should include:

- Information about how you will display your data
- Your five sheets
 - Sheet 1: Brainstorm
 - Sheets 2-4: Initial (Alternative) Designs
 - Sheets 5: Realization (Final Design)

The FdS Methodology: Brainstorm (Sheet 1)

The FdS Methodology: Brainstorm (Sheet 1)

Brainstorming involves five stages:

1. Ideate
2. Filter
3. Categorize
4. Combine & Refine
5. Question

Ideas

Filter

Categorize

Combine & Refine

Question

The FdS Methodology: Brainstorm (Sheet 1)

Stage 1 - Ideate:

- Sketch as many ideas as possible (mini-designs)
- Keep “half an eye” on the goal
- Hold off on criticizing these ideas (you’ll do that later)

The FdS Methodology: Brainstorm (Sheet 1)

Stage 2 - Filter:

- Think about what ideas are suitable to your tasks
- Remove/Delete any duplicate or irrelevant ideas
 - Use a single line to cross through these ideas
- Try to fix any impossible ideas
 - Annotate
- Generate new ideas from the deleted ideas

The FdS Methodology: Brainstorm (Sheet 1)

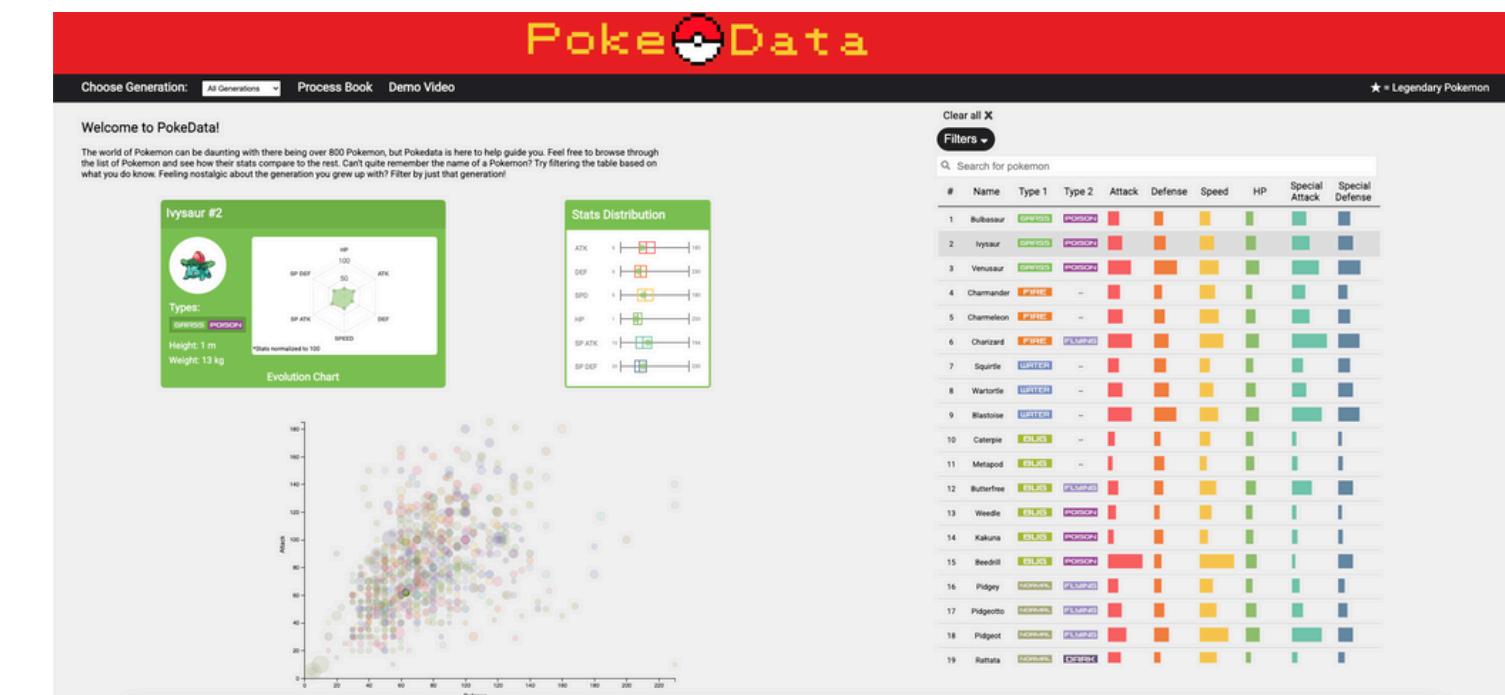
Stage 3 - Categorize:

- Group similar ideas together
- Don't worry about the fine details of categorization
- Consider “what is missing?”
 - Is there another category of designs you should consider?
 - Is this category relevant?

The FdS Methodology: Brainstorm (Sheet 1)

Stage 4 - Combine & Refine:

- Organize your mini-designs into bigger solutions
- Which visualizations complement each other?
- Refine your ideas (make sure they use best practices)
- Combine your visualizations into **Multiple Coordinate Views**
- Start thinking about which combinations you want to plan in more detail in sheets 2, 3, and 4
 - Draw circles around these ideas



The FdS Methodology: Brainstorm (Sheet 1)

Stage 5 - Question:

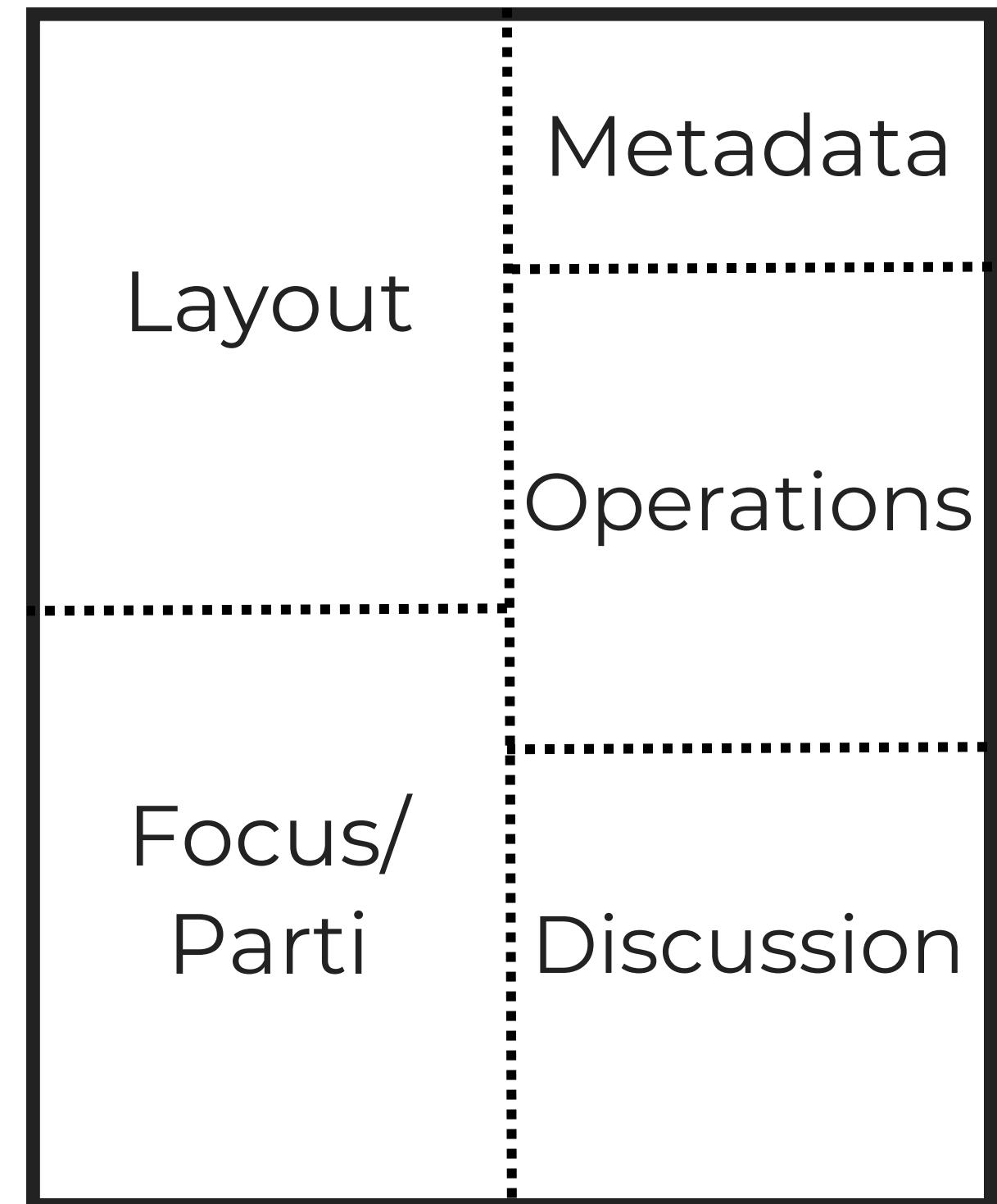
- Reflect on your ideas from stage 4
 - Do your solutions meet the tasks?
 - Are your designs effective?
 - Do your designs mislead or lie?
 - What are the pros and cons of your designs?
 - Do you have three or more distinct designs?

The FdS Methodology: Initial Designs (2, 3, & 4)

The FdS Methodology: Initial Designs (2, 3, & 4)

Five parts to your **alternative** designs:

1. Metadata
2. Layout
3. Operations
4. Focus/Parti
5. Discussion



The FdS Methodology: Initial Designs (2, 3, & 4)

Part 1 - Metadata

- Title, group member names, date, tasks, and sheet #

The FdS Methodology: Initial Designs (2, 3, & 4)

Part 2 - Layout

- A sketch of the interface
 - GUI elements, visualizations, menus, etc.
- Should look like a sketch of a screenshot of the final vis tool

The FdS Methodology: Initial Designs (2, 3, & 4)

Part 3 - Operations

- Action → Result pairs
 - e.g., on result of a button click, load and visualize the data
- Consider what happens when a user interacts with buttons, sliders, lasso selection, and other gestures

The FdS Methodology: Initial Designs (2, 3, & 4)

Part 4 - Focus/Parti

- Parti Pris (French for bias)
 - The central idea / core concept of your visualization
 - E.g., a single core UI component, or a flow chart of the way a user will interact with the data

The FdS Methodology: Initial Designs (2, 3, & 4)

Part 5 - Discussion

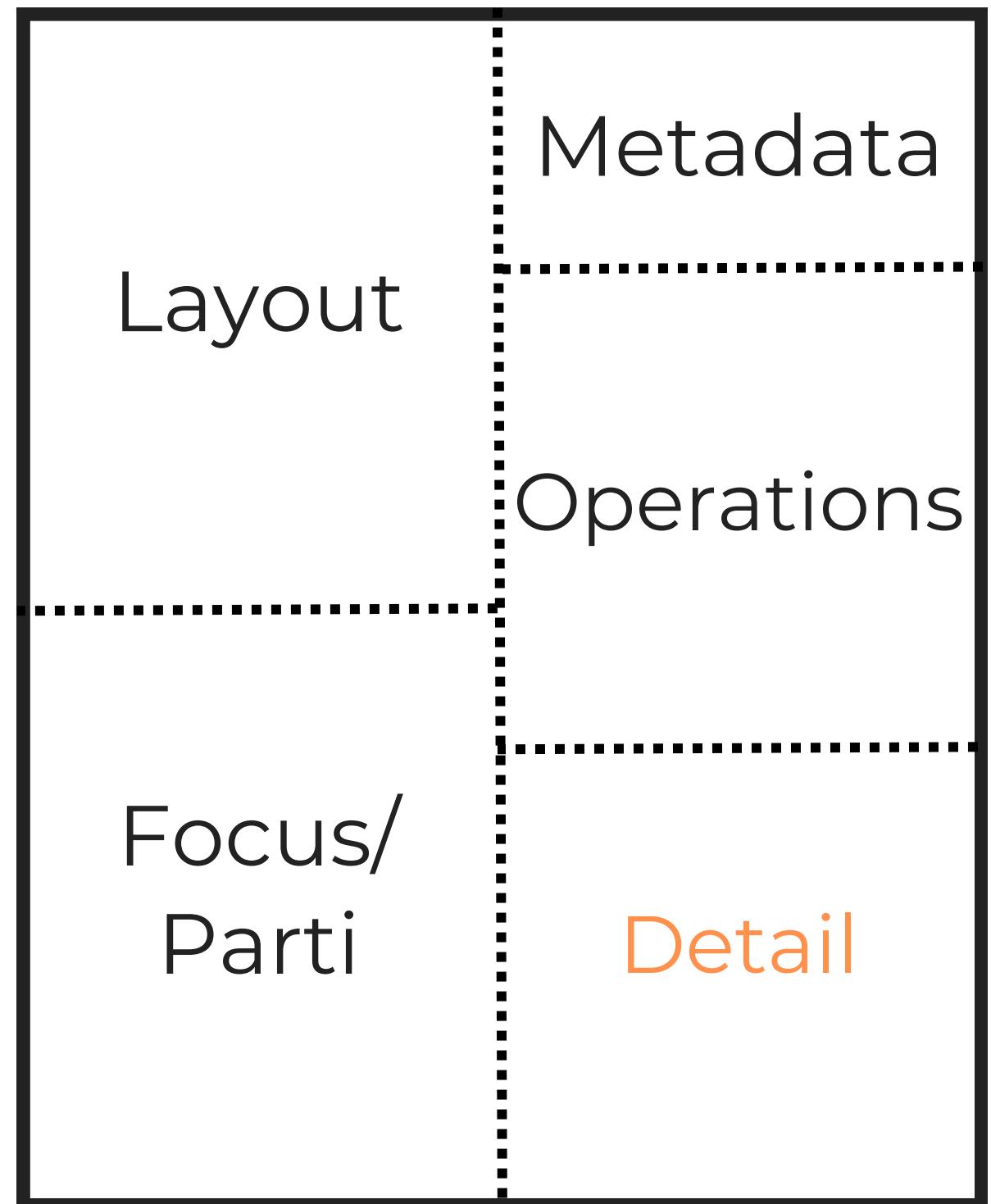
- What are the advantages and disadvantages of this design?
- Suitable to your tasks?
- Feasible to implement? Scalable with more data?
- Is the design clear?

The FdS Methodology: Realization (Sheet 5)

The FdS Methodology: Realization (Sheet 5)

Five parts to your final design:

1. Metadata
2. Layout
3. Operations
4. Focus/Parti
5. Detail



The FdS Methodology: **Realization** (Sheet 5)

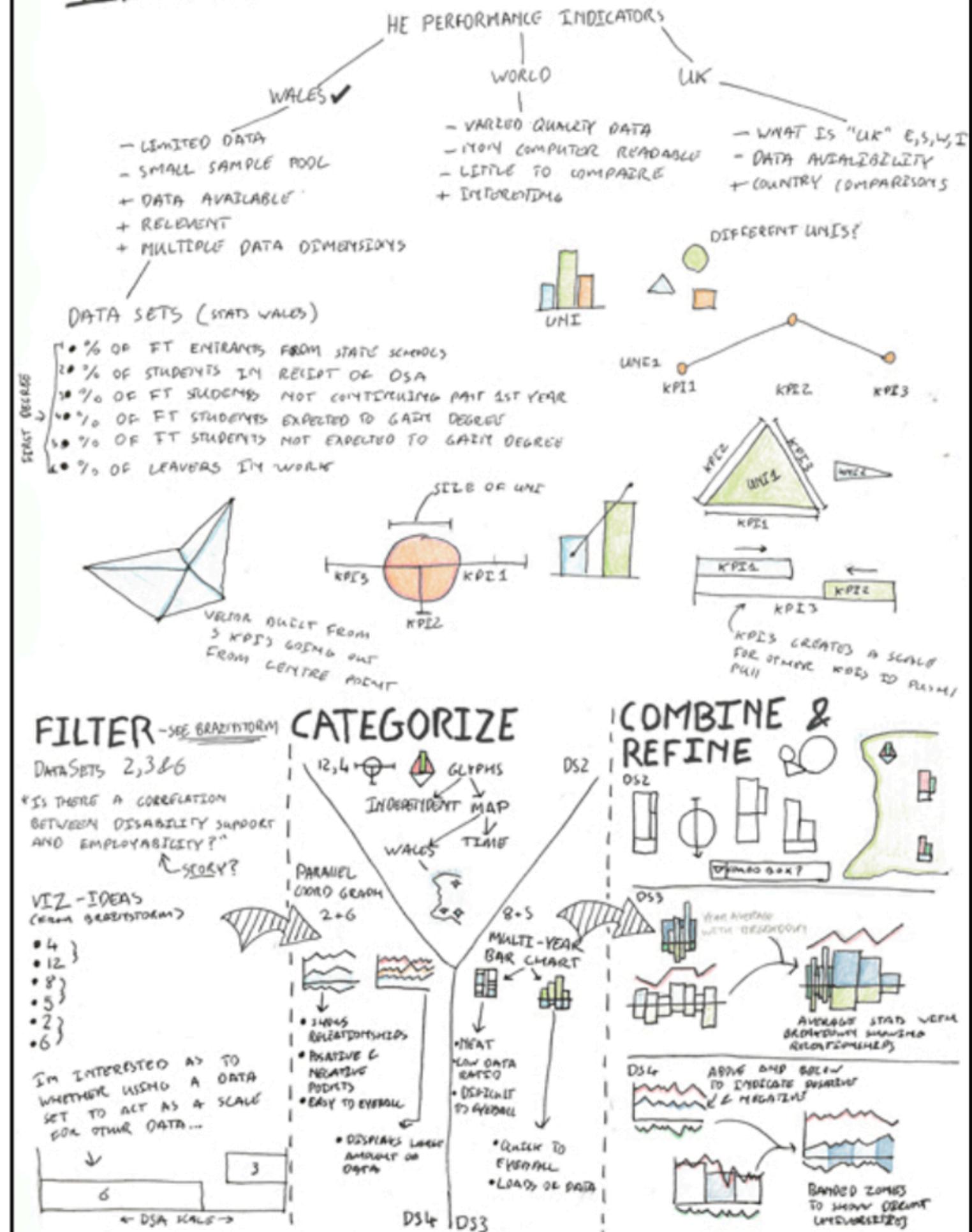
Part 5 - Detail

- Algorithms, data structures, design patterns
- Calculations, UI proportions
- Software dependencies
- Estimate how long it will take to implement
- Any other hardware/software requirements?

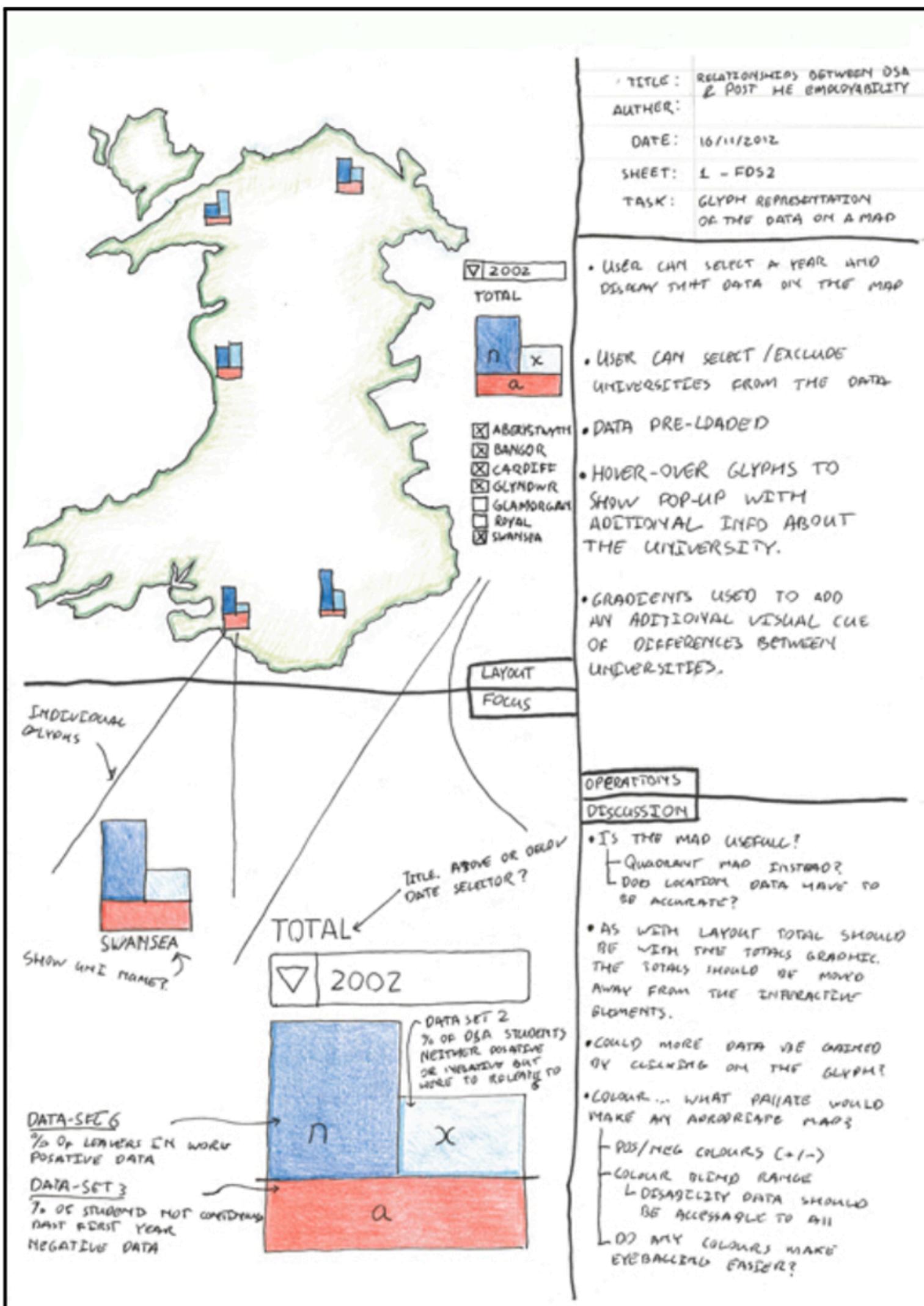
Example: Sheet 1

Data about University Access for disabled students

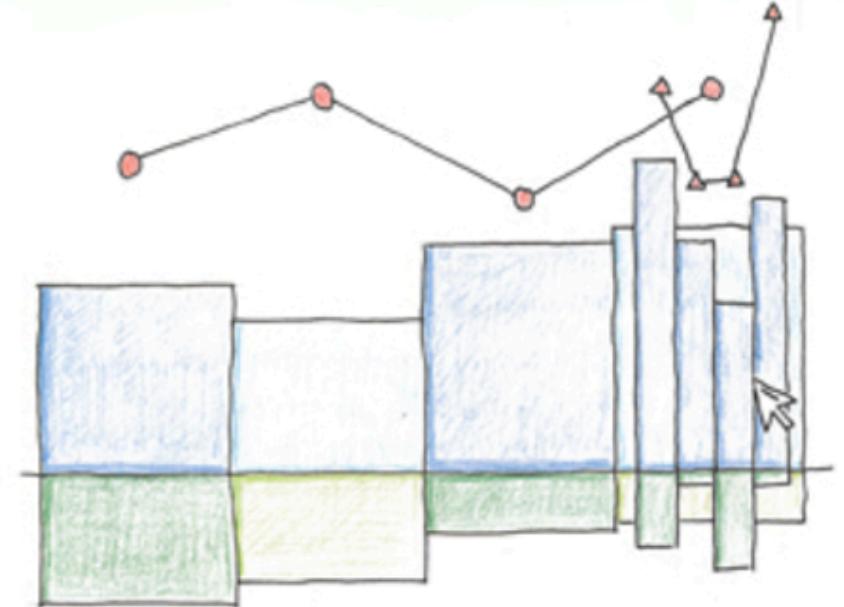
IDEAS



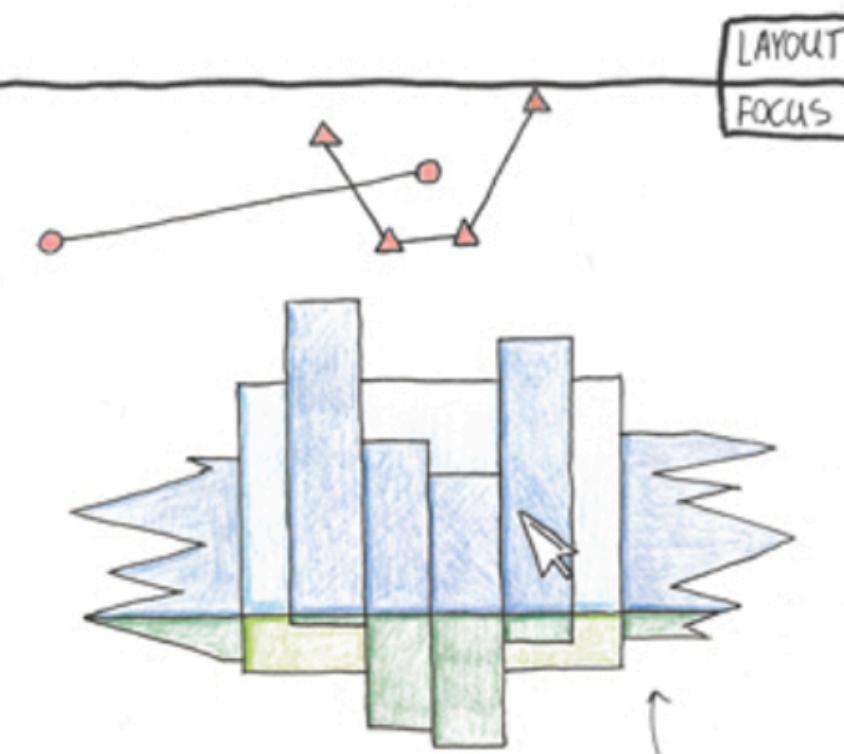
Sheet 2 - Initial Design #1



Sheet 3 - Initial Design #2



UNI-1 ⊖ UNI-2 ⊖ UNI-3 ⊖ UNI-4 ⊖



FOCUS ON THE UNI
THE USER HOVERS
OVER. THIS DISPLAYS
A YEAR BY YEAR
BREAKDOWN.

NOT TO SCALE

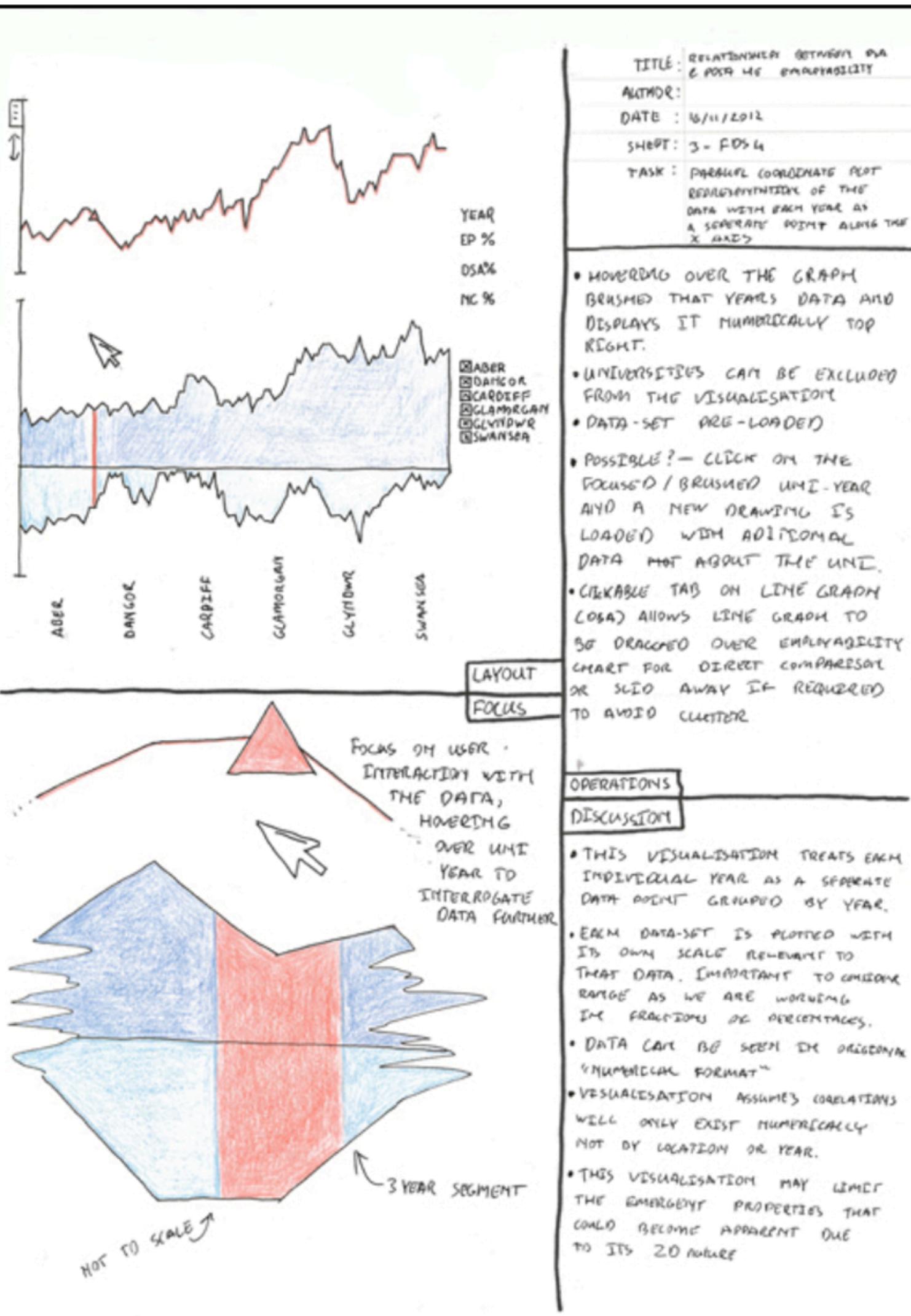
TITLE : RELATIONSHIPS BETWEEN DSA & POST HSC EMPLOYABILITY
AUTHOR:
DATE : 16/11/2012
SHEET : 2 - F053
TASK : BAR-CHART REPRESENTATION
OF THE DSA, EMPLOYABILITY
& LEAVES DATA

- HOVERING OVER EACH AVERAGED UNIVERSITY BAR DRAWS THE YEAR BY YEAR BREAKDOWN.
- CLICKING ON AN INDIVIDUAL YEAR WOULD SET ALL BACKGROUND "MAIN" BARS TO THE FIGURES OF THAT YEAR ALONE FOR COMPARISON.
- CLICKING ON THE MINUS SYMBOL WOULD MINIMISE THAT UNI WITH THE REMAINING UNIS EXPANDING TO FILL THE SPACE.
- DATA PRE-LOADED?
- ABILITY TO CHANGE DISPLAY COLOR COLOURS FOR DIFFERENT USERS PREFERENCES?

**OPERATIONS
DISCUSSION**

- DOES 'MINIMISING' UNI'S ADD ANY USEFUL INTERACTIONS?
- DURING THE HOVER-OVER BREAKDOWN OF DATA SHOULD THE YEAR BE DISPLAYED IN TEXT FOR EASE OF INTERPRETATION
- SHOW ALL BUTTON COULD BE USEFUL BUT IT MAY CLUTTER THE DATA.
- VISUALISATION ASSUMES THE ONLY CORRELATIONS WILL BE NUMERICAL.
- COULD OTHER DATA BE DISPLAYED ON THE BARS? PERCENTAGES OR DEGREE CLASSIFICATION BY TYPE?

Sheet 4 - Initial Design #3



Sheet 5 - Realization

TITLE: RELATIONSHIPS BETWEEN DATA & POST ME EMPLOYABILITY

AUTHOR:

DATE: 17/11/2012

sheet: 4 - FDS 5

task: FINAL DESIGN CONCEPT USING 3 STATES, MAP, QUADRANT MAP & BAR CHART

development: DEVELOPED FROM FDS2, FDS3 & CLIENT COMMUNICATION

STATE 1

STATE 2

STATE 3

CHANGED VIEW

2006

ABERYSTWYTH
BANGOR
CARDIFF
GLAMORGAN
GLYNWYR
SWANSEA
ROYAL

LOCATION & MAP IN STATE 1 FOR VISUAL REFERENCING OF UNIT LOCATION

CHANGED VIEW

2006

ABERYSTWYTH
BANGOR
CARDIFF
GLAMORGAN
GLYNWYR
SWANSEA
ROYAL

SIMPLIFIED LOCATION MAP FOR EASIER INTERLEGACY OF DATA BY LOCATION

RETURN

OPERATIONS

DETAIL

- ALL DATA PROG LOADED
- "SWITCH VIEW" BUTTON ALLOWS USER TO SWITCH BETWEEN STATE 1 & STATE 2
- IN STATE 2 CLICKING ON A LINE GLYPH TRANSITIONS TO STATE 3, A FULL BREAKDOWN OF THE LINE'S DATA BY YEAR.
- IN STATE 3 CLICKING ON THE "RETURN" BUTTON WOULD RETURN THE USER TO STATE 2.
- UNITS CAN BE EXCLUDED/INCLUDED IN THE VIEW (STATES 1&2) BY SELECTING AND DESELECTING THEM.
- YEAR SELECTABLE BY USER

LAYOUT

FOCUS

EMPLOYABILITY

% OF DSA

% OF MAF CONTINUING AFTER 1ST YEAR

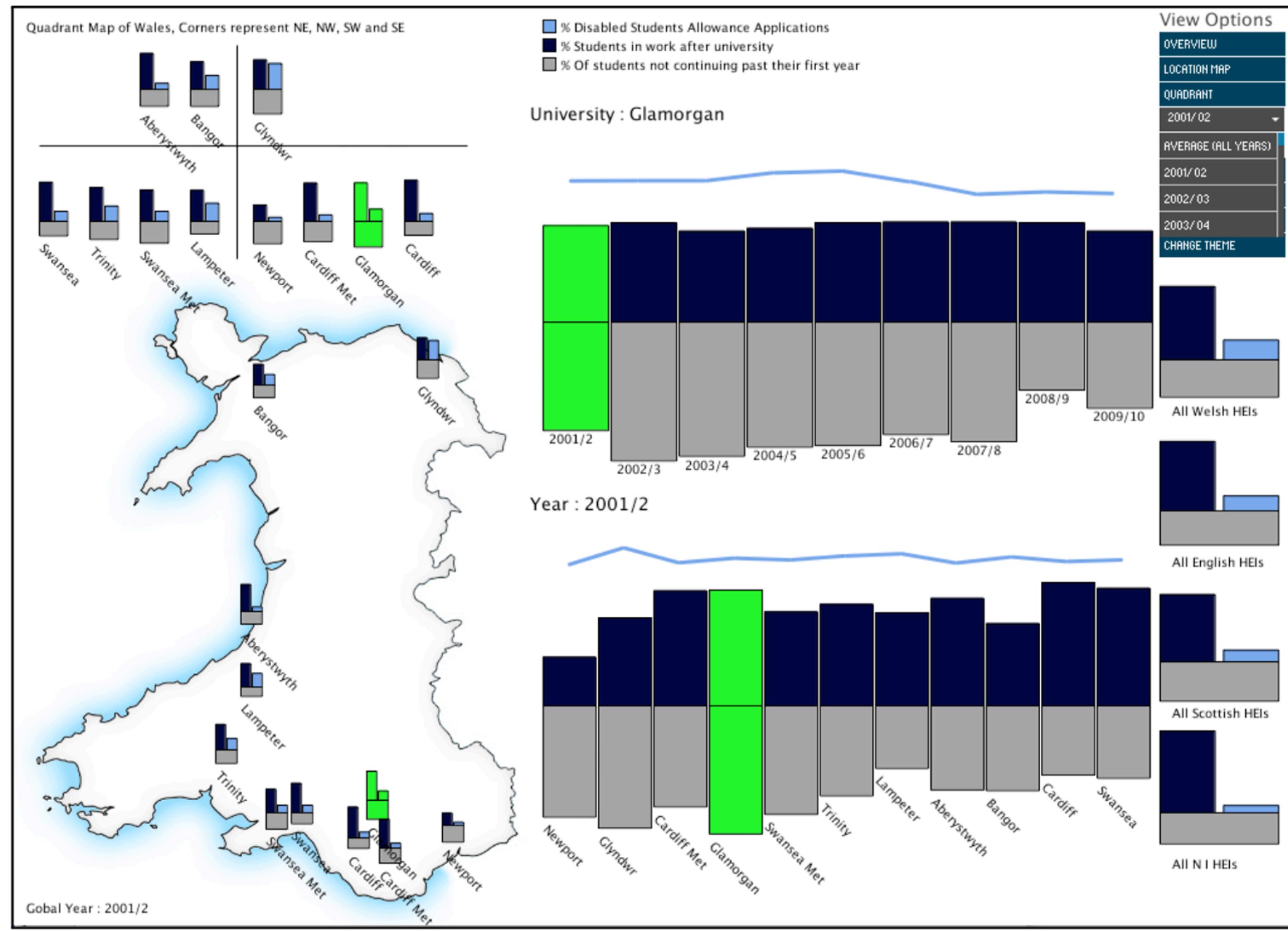
FOCUS WILL CHANGE DEPENDING ON CURRENT VIEW STATE. MAIN FOCUS WILL BE ON THE BI-POLAR GLYPH (RIGHT) WITH POSITIVE DATA ABOVE THE DATUM AND NEGATIVE BELOW THE DATUM. THE SCALE WILL BE DIFFERENT (NUMERICALLY) FOR EACH DATA ITEM NUMBER WILL HAVE THE SAME UPPER AND LOWER LIMITS VISUALLY..

OPERATIONS

DETAIL

- TIME TO BUILD ESTIMATED AT 16 HOURS
- DATA-SETS ACQUIRED FROM STATE WALES
- SCALE FOR EACH ITEM SET TO THE RANGE AS THE DIFFERENCE IS FRACTIONS OF %'s.
- COLOUR CHOICE BASED ON SURVEY DATA, HOWEVER WOULD NOT BE SCIENTIFIC FOR COLOUR BLIND. MAY CHOOSE YELLOW, PURPLE & BLUE TO SUIT DEUTERANOMALY & PROTANOMALY COLOUR BLINDNESS.
#DPSAAB - BLUE,
#EDE544 - YELLOW,
#B33352 - PURPLE.
- DATA-SET WILL NEED SOME CLEANING AS NOT ALL UNITS HAVE ALL YEARS DATA.
- MAY NOT INCLUDE STATE THREE L STATE THREE BASED ON FDS3.

Final Website



Your Project Proposal Will Include:

6. Visualization Design (Five Design Sheet Methodology)

This section should include:

- Information about how you will display your data
- Your five sheets
 - Sheet 1: Brainstorm
 - Sheets 2-4: Initial (Alternative) Designs
 - Sheets 5: Realization (Final Design)

Your Project Proposal Will Include:

- 1. Project Metadata**
- 2. Background & Motivation**
- 3. Project Objectives**
- 4. Data**
- 5. Data Processing**
- 6. Visualization Design (Five Design Sheet Methodology)**
- 7. Must-Have Features**
- 8. Optional Features**
- 9. Project Schedule**

Your Project Proposal Will Include:

7. Must-Have Features

This section should include:

- A list of features without which you would consider your project to have failed

Your Project Proposal Will Include:

- 1. Project Metadata**
- 2. Background & Motivation**
- 3. Project Objectives**
- 4. Data**
- 5. Data Processing**
- 6. Visualization Design (Five Design Sheet Methodology)**
- 7. Must-Have Features**
- 8. Optional Features**
- 9. Project Schedule**

Your Project Proposal Will Include:

8. Optional Features

This section should include:

- A list of features that would be nice to have, but aren't critical to your project

Your Project Proposal Will Include:

- 1. Project Metadata**
- 2. Background & Motivation**
- 3. Project Objectives**
- 4. Data**
- 5. Data Processing**
- 6. Visualization Design (Five Design Sheet Methodology)**
- 7. Must-Have Features**
- 8. Optional Features**
- 9. Project Schedule**

Your Project Proposal Will Include:

9. Project Schedule

This section should include:

- A list of weekly deadlines for the remainder of the semester
- Details about which group member will complete which tasks

Managing Your Projects with Git

Git is a distributed **version-control** system

Note: Git refers to a “version” as a “commit”

Git keeps track of the history of your commits:

- You can go back and look at earlier versions
- Or even give up on the current version and roll back to an earlier version

Git is adaptable to a variety of workflows

Git is a **distributed** version-control system

You keep your files in a “repository” on your local machine

- This is basically just the folder where your project code is located

As you make changes to your code, you can synchronize your **local repository** with a **remote repository (on GitHub)** on a server

- This way, if your computer crashes or dies, **your code is protected** on the server
- If you **move to a different machine**, you can pick up the changes by synchronizing the new machine with the server
- If you **work on a team**, other people’s versions can be synchronized using the server

Git Tools and Commands

Git is very flexible and has many tools for you to use

You can do a lot with Git

- Even accidentally destroy your entire commit history
- It's very important to understand how the underlying Git model works

Git Tools and Commands

Setup a new repository, or create/manage a new branch to test out an idea

- `init`, `checkout`, `branch`, `clone`

Make modifications to the local or remote repository

- `add`, `commit`

Get information about your working directory or commits

- `status`, `diff`, `log`

Create reference points

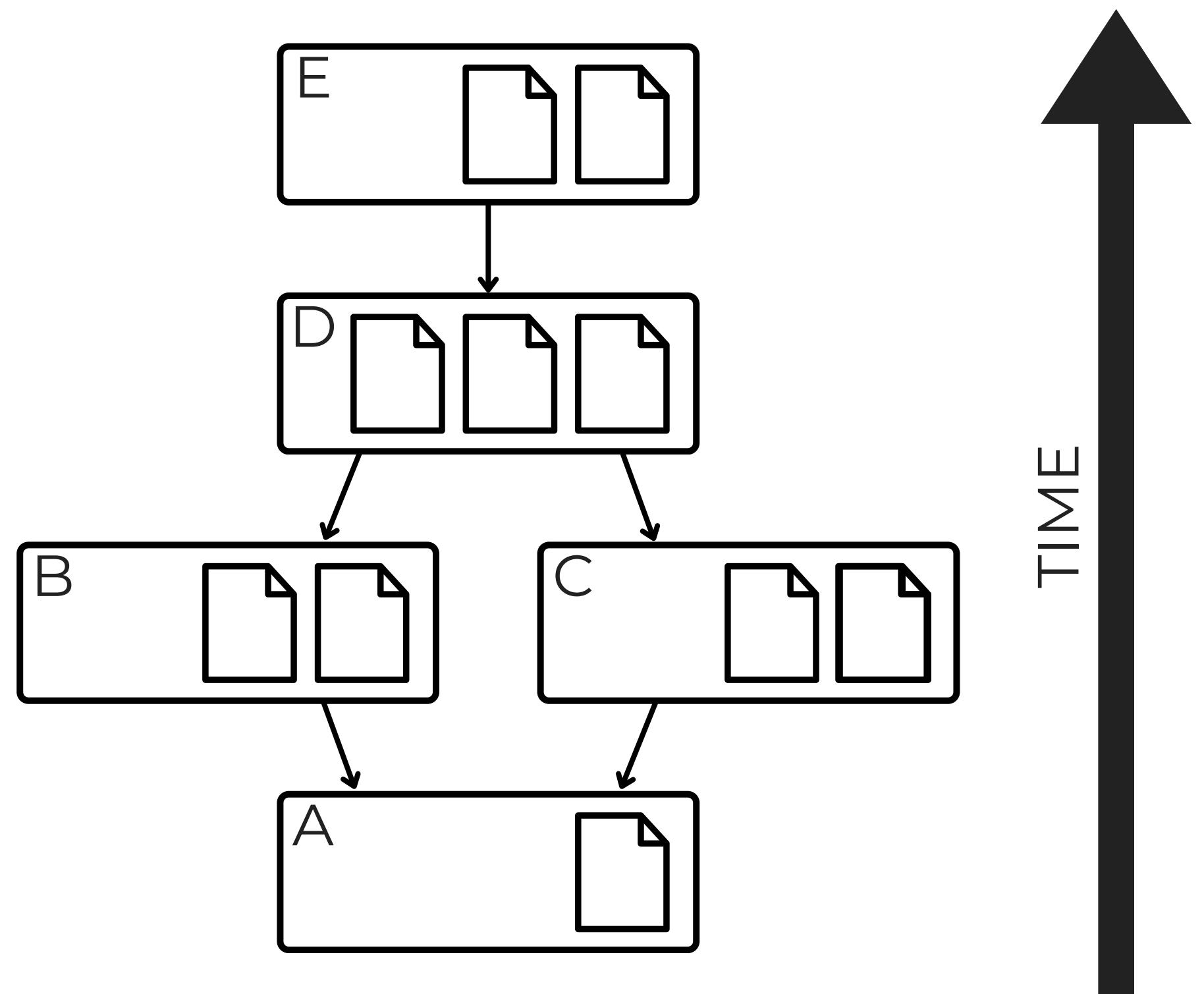
- `tag`, `branch`

Synchronize with remote repository

- `push`, `pull`, `fetch`, `sync`

Your Git Repository Contains:

- Files & Directories
- Commits
- Ancestry Relationships



The Git Ancestry Graph:

Commits: snapshots of file status

Tags:

- identify versions of interest
- e.g., “releases”

Branches:

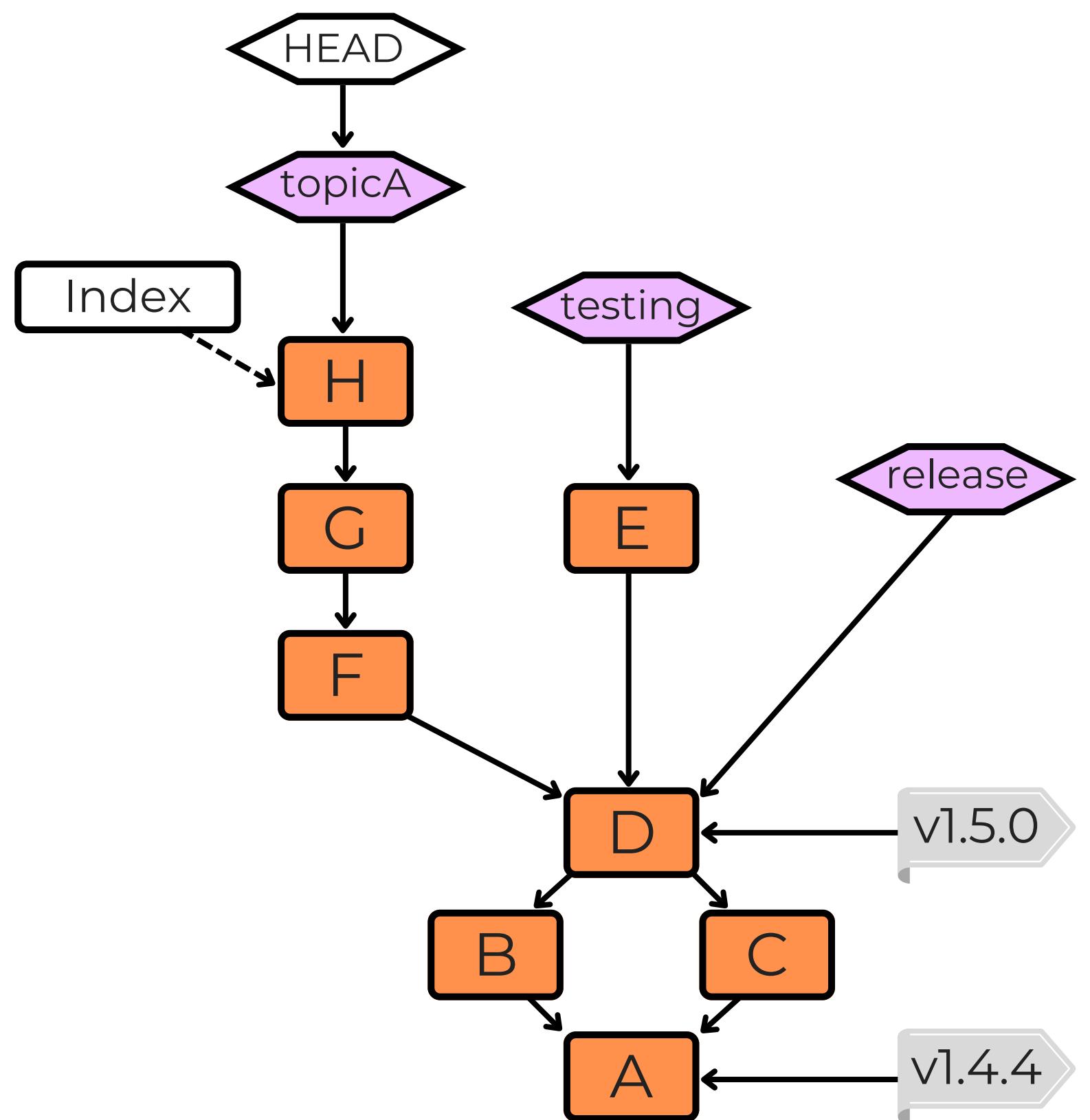
- Divergent path for source code modification

HEAD:

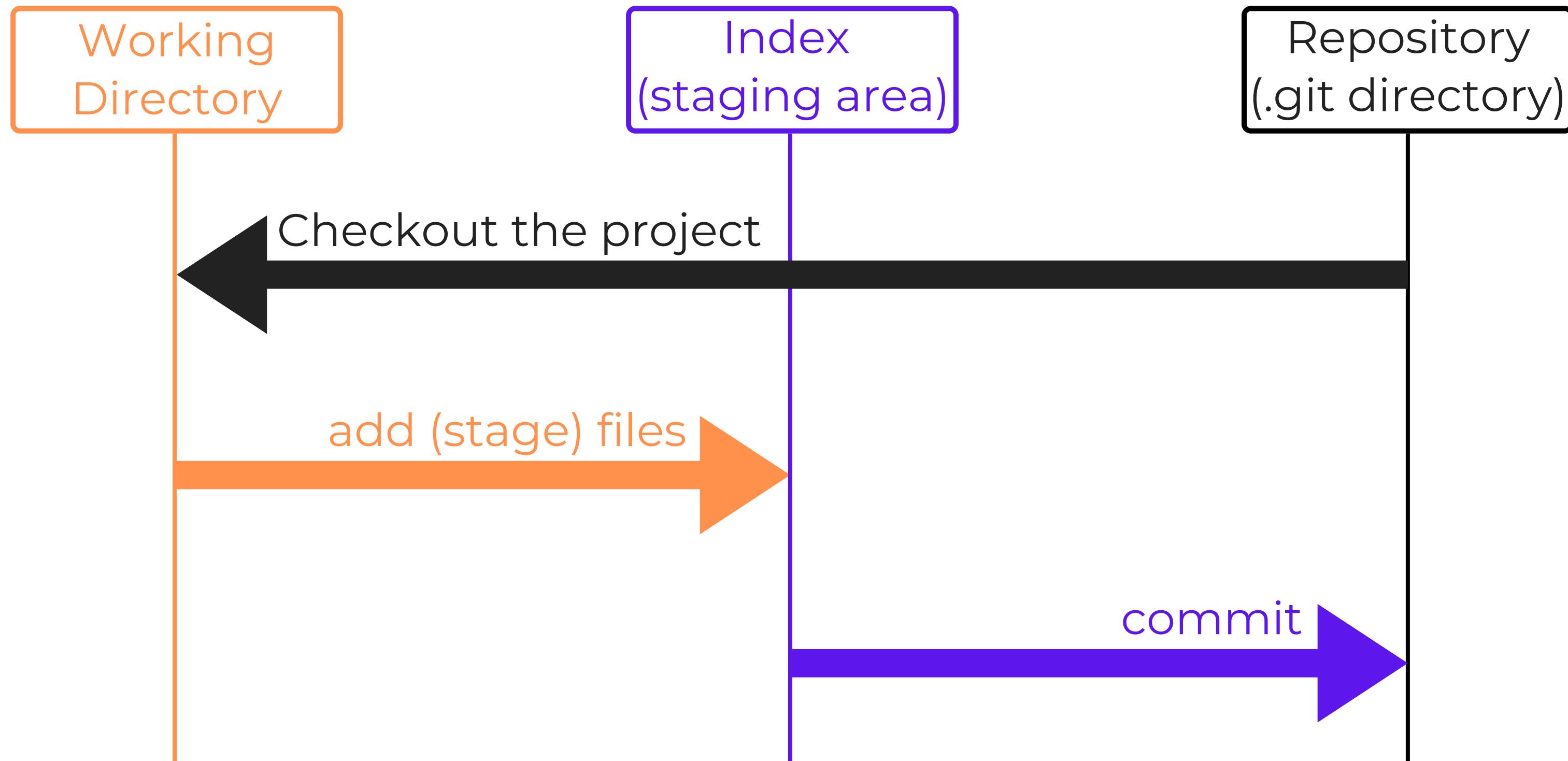
- current checkout
- usually points to a branch

Index: “staging area”, to be committed

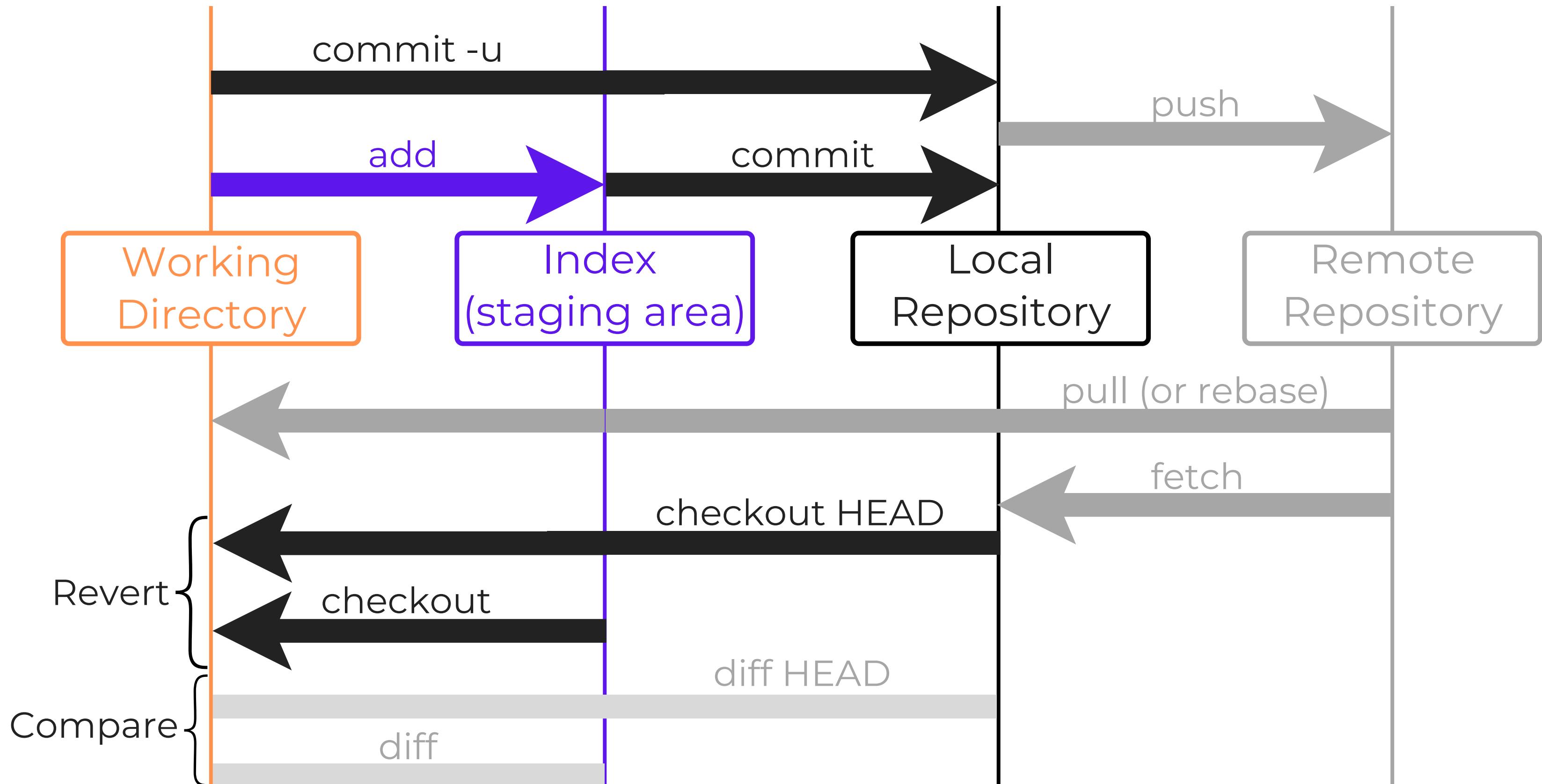
Directed Acyclic Graph (DAG)



Local Git Operations



Git Transport Commands



Git Software

Windows:

- Git command line tools - <https://git-scm.com/download/win>
- Git GUI - <https://tortoisegit.org/>

Mac:

- Several ways to install git
 - (see <https://git-scm.com/downloads/mac>)

Linux:

- git should already be installed. If not, use the appropriate package manager (apt or yum) to install it

Getting Started

- Create a GitHub account, if you don't already have one
 - <https://github.com/>
- Setup a repository on GitHub for your project
- Once you have created a remote repository
 - determine the remote path of the repository, and
 - choose a local working directory for your code
- Use `git clone` to sync your remote repository to the local working directory

Finding the Remote Path

A screenshot of a GitHub repository page for 'FastGPT'. The page shows the repository's main branch, 9 branches, and 175 tags. A red arrow points to the 'Star' button, which has 20.9k stars. The repository has 130 watchers and 5.4k forks. The 'About' section describes FastGPT as a knowledge-based platform built on LLMs, offering capabilities like data processing, RAG retrieval, and visual AI workflow orchestration. The repository's history shows several commits from 'c121914yu' and other contributors.

labring / FastGPT

Type / to search

Code Issues 350 Pull requests 3 Discussions Actions Security Insights

FastGPT Public Watch 130 5.4k Star 20.9k

main 9 Branches 175 Tags Go to file Add file Code

c121914yu 4.9.0 test (#3779) 7aacce8 · 8 hours ago 1,773 Commits

.github readme (#3687) last week

.husky perf: ui 2 years ago

.vscode V4.8.14 dev (#3234) 3 months ago

bin fix: the helm release failed due to version handle (#2199) 7 months ago

docSite 4.9.0 test (#3779) 8 hours ago

files 4.9.0 test (#3779) 8 hours ago

packages perf: unlogin user fetch data (#3775) 11 hours ago

projects 4.9.0 test (#3779) 8 hours ago

python marker doc (#3335) 2 months ago

scripts V4.8.15 feature (#3331) 2 months ago

.dockerignore 4.6.5- CoreferenceResolution Module (#631) 2 years ago

About

FastGPT is a knowledge-based platform built on the LLMs, offers a comprehensive suite of out-of-the-box capabilities such as data processing, RAG retrieval, and visual AI workflow orchestration, letting you easily develop and deploy complex question-answering systems without the need for extensive setup or configuration.

tryfastgpt.ai

react workflow nextjs openai
claude rag gpt-4 llm qwen
deepseek

Readme View license Activity Custom properties 20.9k stars

Finding the Remote Path

The screenshot shows a GitHub repository page for 'FastGPT' owned by 'labring'. The repository is public and has 350 issues, 3 pull requests, and 20.9k stars. The 'Code' tab is selected. On the right, there's an 'About' section describing FastGPT as a knowledge-based platform built on LLMs, offering data processing, RAG retrieval, and visual AI workflow orchestration. Below the description are buttons for 'tryfastgpt.ai' and various tags like react, workflow, nextjs, openai, claude, rag, gpt-4, llm, qwen, and deepseek. A red box highlights the 'Clone' section where the GitHub CLI URL 'git@github.com:labring/FastGPT.git' is listed under the 'HTTPS' tab.

labring / FastGPT

Type / to search

Code Issues 350 Pull requests 3 Discussions Actions Security Insights

FastGPT Public

Watch 130 Fork 5.4k Star 20.9k

main 9 Branches 175 Tags

Go to file Add file Code

Local Codespaces

Clone

HTTPS SSH GitHub CLI

git@github.com:labring/FastGPT.git

Use a password-protected SSH key.

Open with GitHub Desktop

Download ZIP

11 hours ago 8 hours ago 2 months ago 2 months ago 2 years ago

Readme View license Activity Custom properties 20.9k stars

About

FastGPT is a knowledge-based platform built on the LLMs, offers a comprehensive suite of out-of-the-box capabilities such as data processing, RAG retrieval, and visual AI workflow orchestration, letting you easily develop and deploy complex question-answering systems without the need for extensive setup or configuration.

tryfastgpt.ai

react workflow nextjs openai
claude rag gpt-4 llm qwen
deepseek

Sample Session Commands

```
> git clone <remote_path> <local_directory>
> cd <local_directory>
> git pull
> touch newfile.txt
> git add newfile.txt
> git commit -m "added a new file"
> git push
```

Recommendations for Your Workflow

- Commit regularly.
 - After you make new changes, it's a good idea to commit
- Push to the remote repository at the end of every work session (or if you need to switch computers)

Useful References for Git

- <https://book.git-scm.com/index.html>
- <https://learngitbranching.js.org/>
- <https://www-cs-students.stanford.edu/~blynn/gitmagic/>
- So many YouTube videos:
 - e.g., <https://www.youtube.com/watch?v=HVsySz-h9r4>

FIN

Upcoming Dates

Feb 13 (Today): Announce Your Project

Feb 14 (Tomorrow): ~~Group~~ Activity 1 Due

Feb 14 (Tomorrow): Homework 1 Due

Feb 28: Project Proposal Due