

PV160

HCI Lab + Visit Lab

List of topics, spring 2021

Spectatorship Experience

Spectating other people playing games has become a mass phenomenon over the last years

This can range from small gatherings around the family table following a card or board game to large e-sport events attracting up to millions of viewers

This raises new design challenges for HCI:

- How can spectator experiences be facilitated in order to pro-actively involve spectators instead of being passive observers?
- How can an approach look like to make games more accessible for the audience, to spark interest, and to stimulate a feeling of togetherness?
- <https://seegamesws.wordpress.com/>
- <https://esportshci.wordpress.com/>



In-Game Visualization

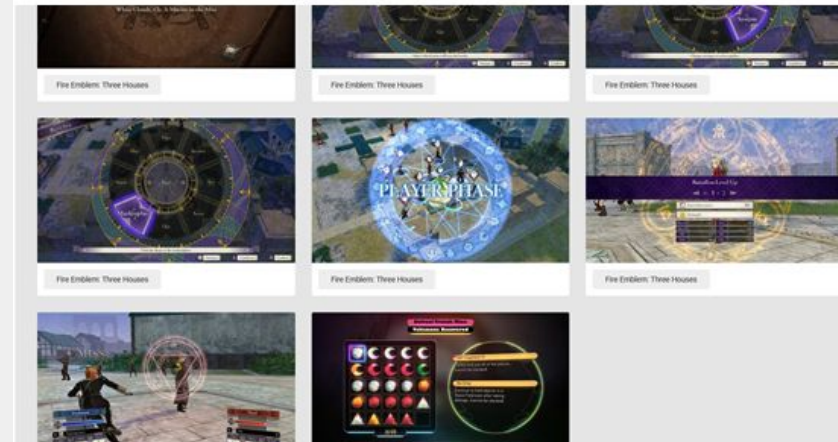
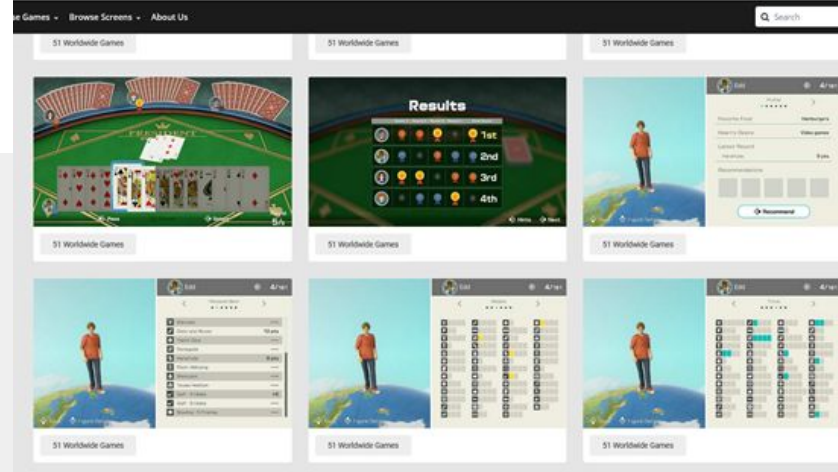
Visualizations in games become more and more important in order to give feedback and as UI element

The [Game UI Database](#) was developed as a free resource for UI/UX designers in the games industry with the goal

- to discovery new inspiration
- to provide reference materials

Goal:

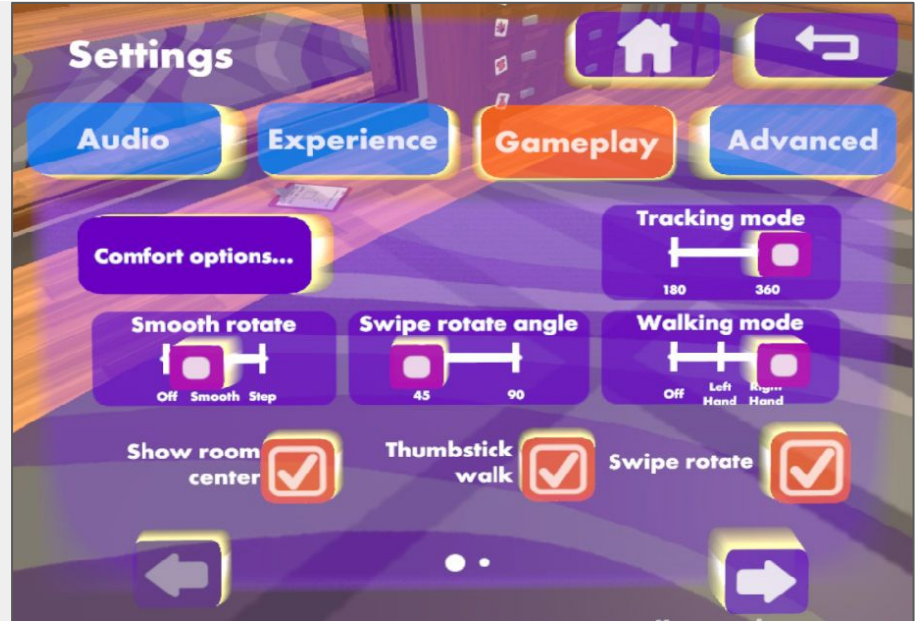
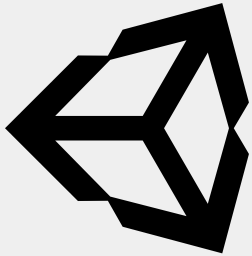
- To analyze the Game UI database regarding visualizations which are used as part of game interfaces
- The different games and screenshots in the database should be categorized according to the used visualizations
- The visualizations should be analyzed with respect to different criteria such as purpose, genre, tasks, year, and others
- This is part of an open master topic “[In-Game Visualization Browser](#)“



Contact - Simone

User Interface for Virtual Reality

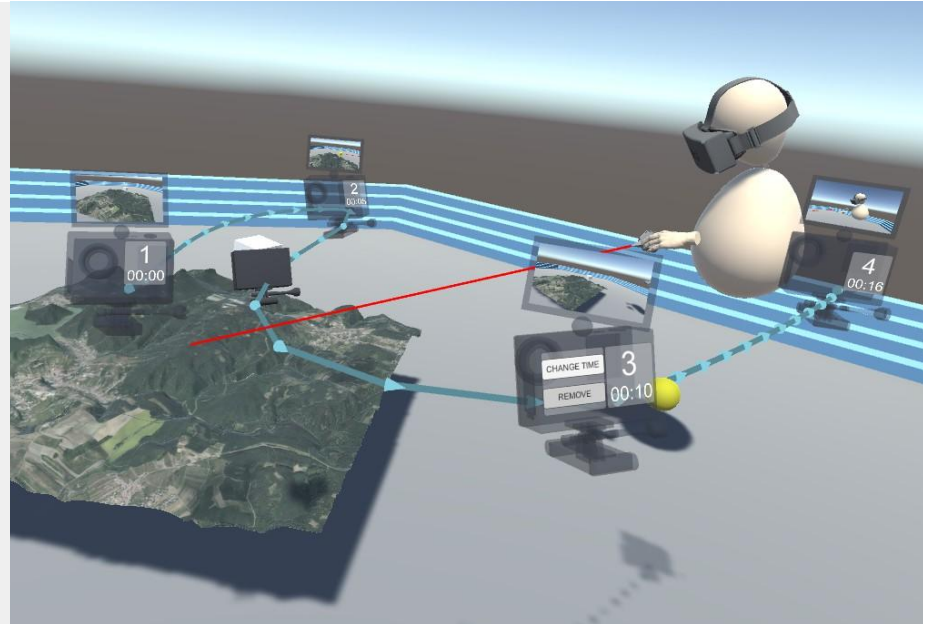
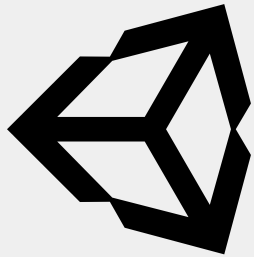
- Universal UI for VR Apps
- Interactive:
 - “Physically” & via pointer
- **Unity**



Contact: Vojta

Interactable Trajectory Visualization in Virtual Reality

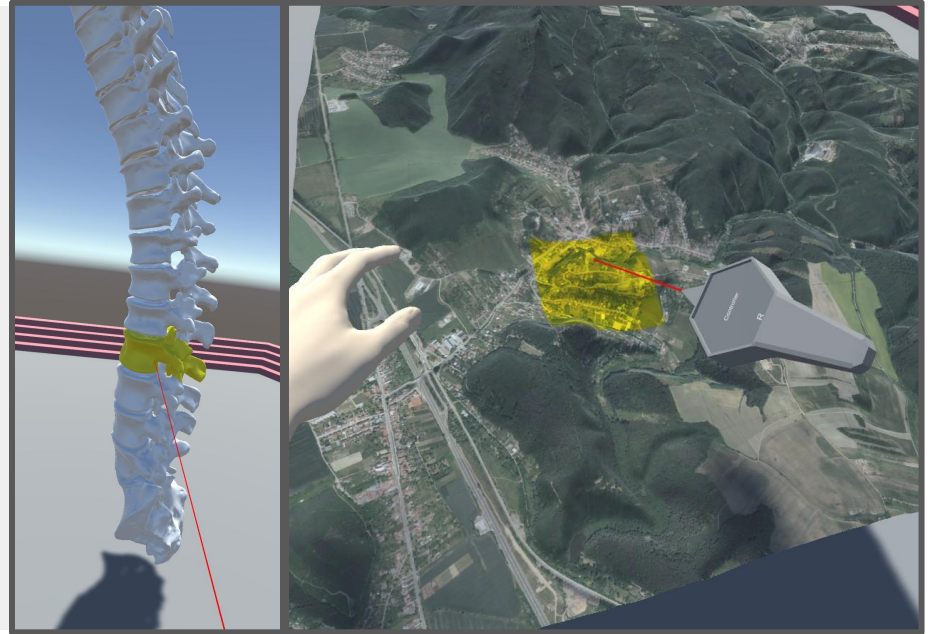
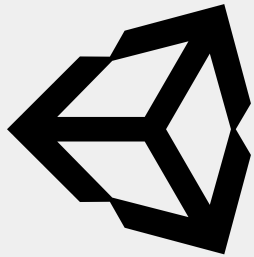
- Be able to **create** and **edit** at runtime
- **Unity**



Contact: Vojta

Objects Grouping and Highlighting in VRdeo

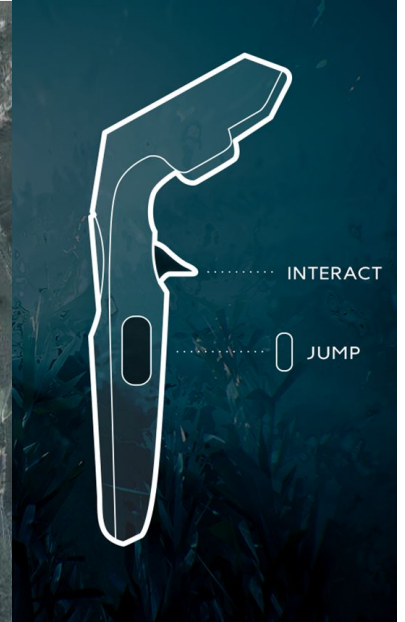
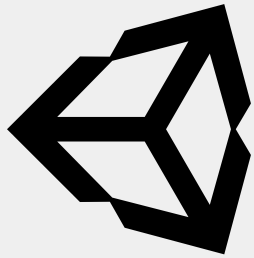
- Group and ungroup objects
- Highlight in 3D
- Unity



Contact: Vojta

Custom Interactive Models of Virtual Reality Controllers

- Highlight individual buttons
- Labels (“Tutorial”)
- Unity



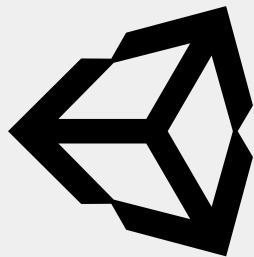
Contact: Vojta

VR Paraglide

- a) simple “physics” of paraglide
- b) “flythrough” - log, analysis, ...

+ Basic controls in VR

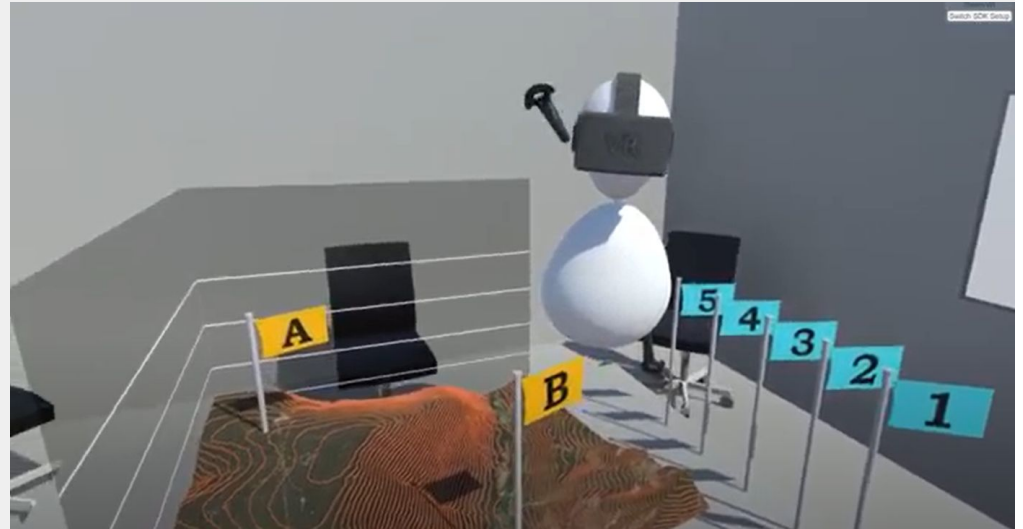
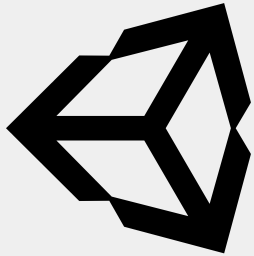
Suitable for bachelor thesis



Contact: Jirka, Vojta

Collaborative VR

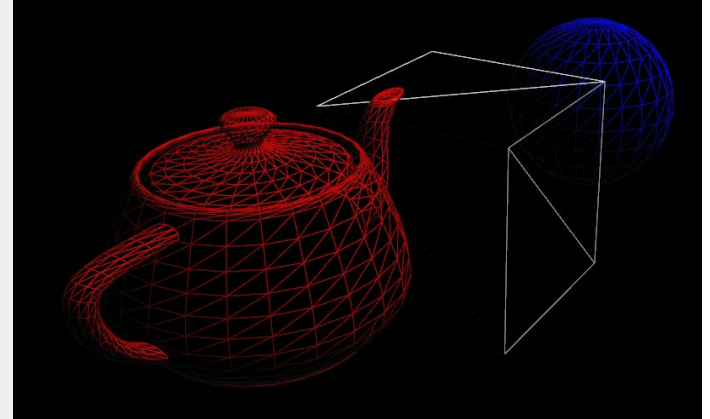
- Various topics related to **networking** and VR
 - Related to voice chat, e.g. speaker highlighting
 - ...



Contact: Jirka, Vojta

E-Learning

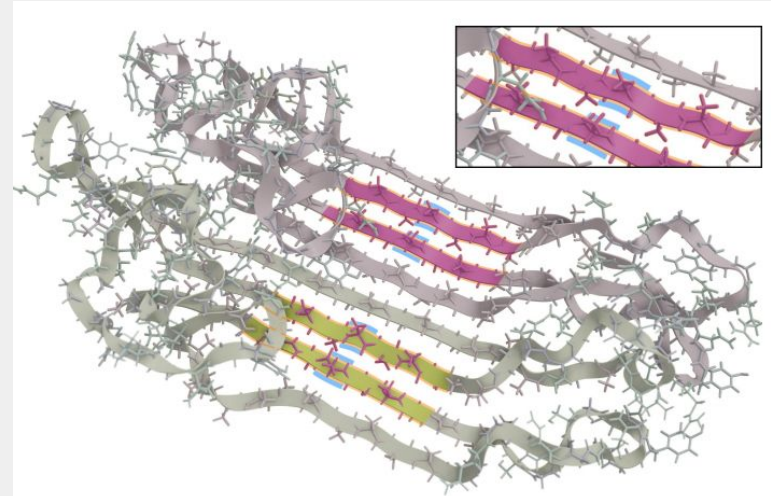
- System for automatic correction of short PB009 programming assignments
- Suitable for bachelor/master thesis



Contact: Jan

Selection Schemes for CAVER Analyst

- Reimplementation of techniques presented in
 - MolFind – Integrated Multi-Selection Schemes for Complex Molecular Structures



Contact: Jan

Filtering UI

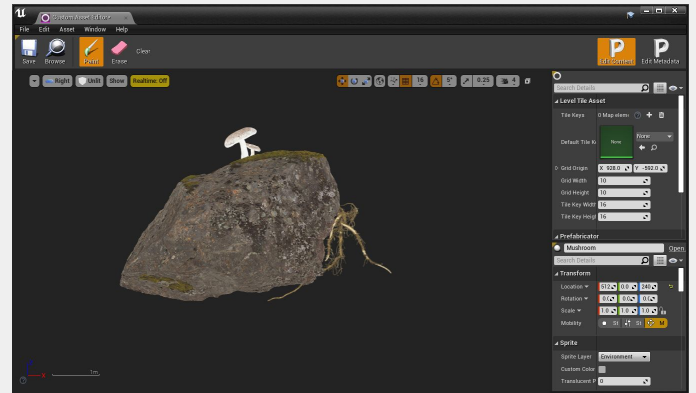
- Implementation of UI for filtering panels in CAVER Analyst



Contact: Jan

Custom Editor for Prefabricator in UE4

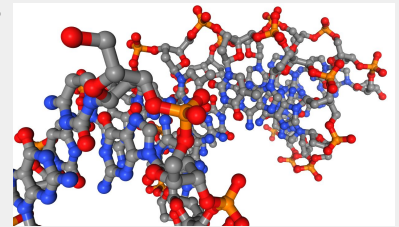
- Edit Prefabricator assets in a separate window
 - experience how tools programming in gamedev works
 - sadly no documentation
 - good understanding of C++ required
 - <https://prefabricator.dev/>



Contact: Jan

Ball-and-stick billboarding in Unity URP

- **Task:** Implement billboarded ball-and-stick model for real-time rendering of large molecules
- **Technology:** Unity, Universal Render Pipeline
- **Deliverables:**
 - Unity project presenting an extensible & well-documented algorithm(s) able to render a large molecule with a ball-and-stick model:
 - using a 3D geometry,
 - Can be unoptimized, primarily for reference.
 - using billboards / imposter rendering.
 - Should have a good performance.



Contact: David

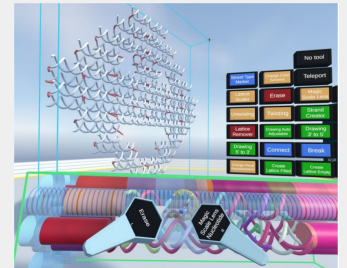
Exploring VR on the web

- **Task:** Examine WebXR standard and relevant frameworks for development of web-based VR apps
- **Technology:** JavaScript
- **Deliverables:**
 - Document / mind map summarizing existing frameworks, which can provide some kind of VR experience via a web browser
 - Each framework above will be accompanied by a description of its capabilities, links to useful tutorials or documentation
 - A working example of a web-based VR scene allowing the user to move around and interact with some of the things in the scene



Vivern: extending the capabilities

- **Task:** Extend Vivern (VR application for DNA origami modeling) with new features
 - Exact features to be agreed on with the supervisor
 - Suggestions:
 - Inter-lattice connections
 - Scaffold routing / auto stapling
 - Cadnano export
- **Technology:** Unity, C# [# sign in this font is really bad :D]
- **Deliverables:**
 - Vivern application with selected features implemented



Contact: David

Modeling a hillclimb road course / race track for Assetto Corsa racing simulator

- **Task:** Create a playable race track for Assetto Corsa game
 - Real-world inspired course is preferred
 - In case of selecting hillclimb road course suggested by supervisor, GPS recordings and real photos can be provided
- **Technology:** any 3D graphics editor, official game editor
- **Deliverables:**
 - 3D model of a racetrack and surroundings
 - Racetrack data which can be directly imported into the game and played without major issues

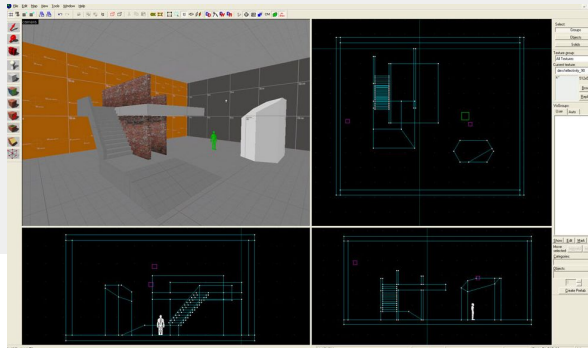


Contact: David

Creating a fully playable Left 4 Dead 2 level

- **Task:** Create a playable level/mission for Left 4 Dead 2
 - Covers all tasks from assembling a level (from own or existing assets), through placement of enemies and items up to enriching the level with custom music
- **Technology:** Left 4 Dead Authoring Tools
- **Deliverables:**
 - Mission (and relevant data) which can be imported into the game and played

LEFT 4 DEAD



Contact: David

(Alpha)numeric input in VR

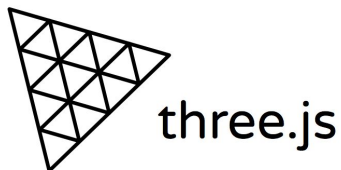
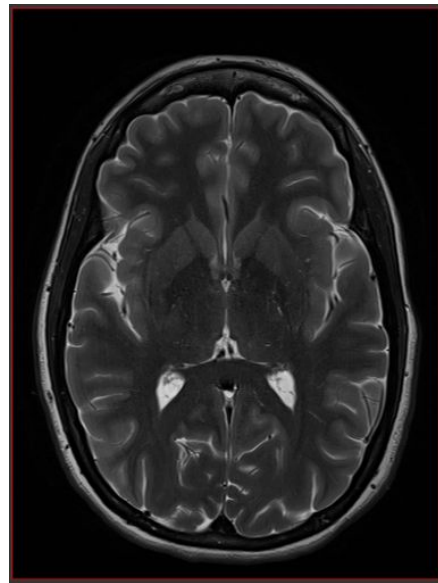
- **Task:** Design and implement several approaches for (alpha)numeric input in VR
- **Technology:** Unity
- **Deliverables:**
 - Results of research of existing approaches
 - Unity VR project with a simple scene containing a text field for the user input
 - At least 5 different ways how to input purely numeric values
 - Numeric VR keyboard, touchpad-controlled “menu”, ???
 - At least 5 different ways how to input alphanumeric values
 - VR keyboard, speech-to-text, ???
 - Physical keyboard of the computer does not count :-)



Contact: David

Components for visualization of MRI data

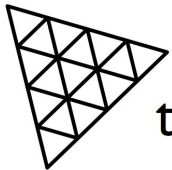
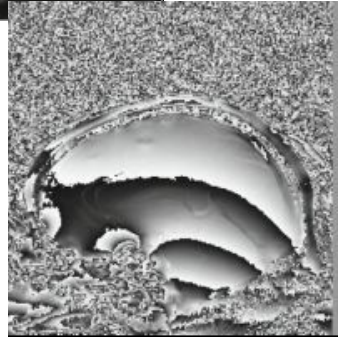
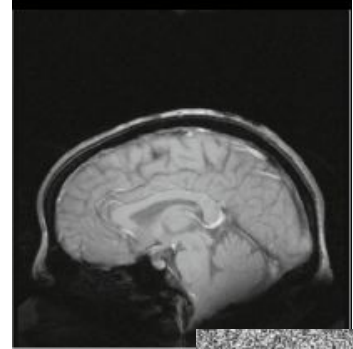
- **Implement one of the following:**
 - Rulers and orientation tools
 - Selection interface
 - Image enhancement interface
- **Technology:** Three.js



Contact: Tomáš

Visualization of complex-valued images

- **Task:**
 - Explore possibilities of visualization of complex-valued images
- **Technology:** Three.js

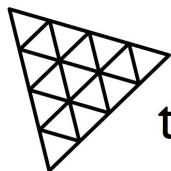
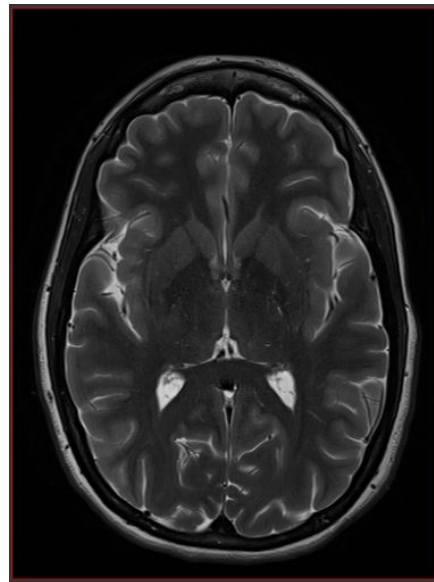


three.js

Contact: Tomáš

Medical data visualization using Electron

- **Task:** Create desktop apps out of some of Three.js examples and evaluate performance
- **Technology:** Electron, Three.js
- **Deliverables:**
 - Compiled applications
 - Performance report

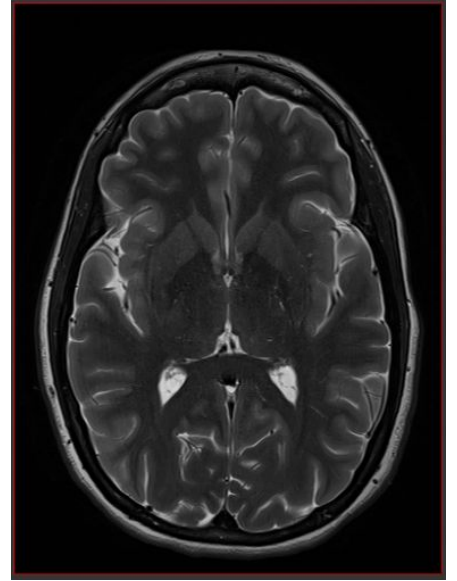
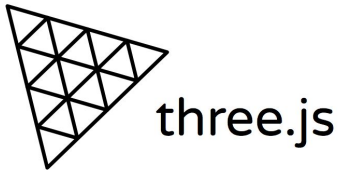


three.js

Contact: Tomáš

Tracking user interactions in AMI

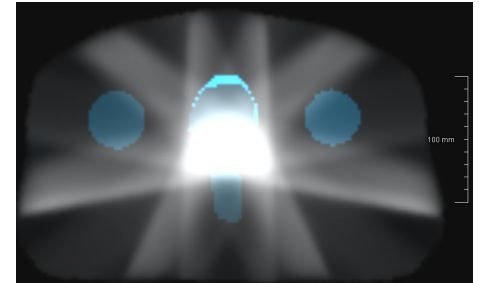
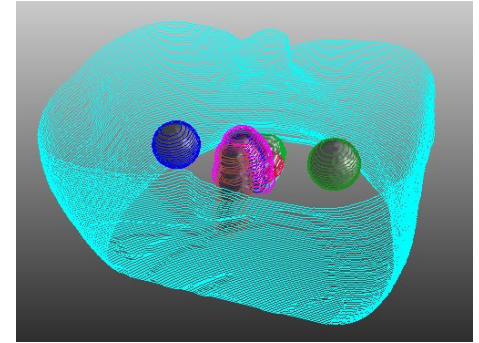
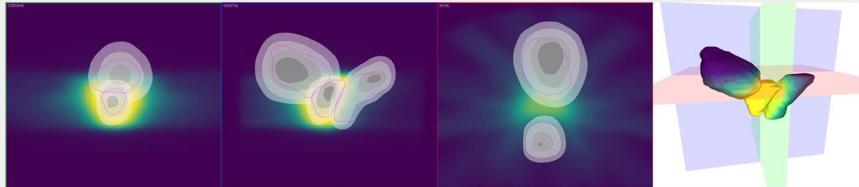
- **Task:** Implement a system to log information about user interaction in the Three.js library
- **Technology:** AMI, Three.js



Contact: Tomáš

Import radiotherapy data in DICOM

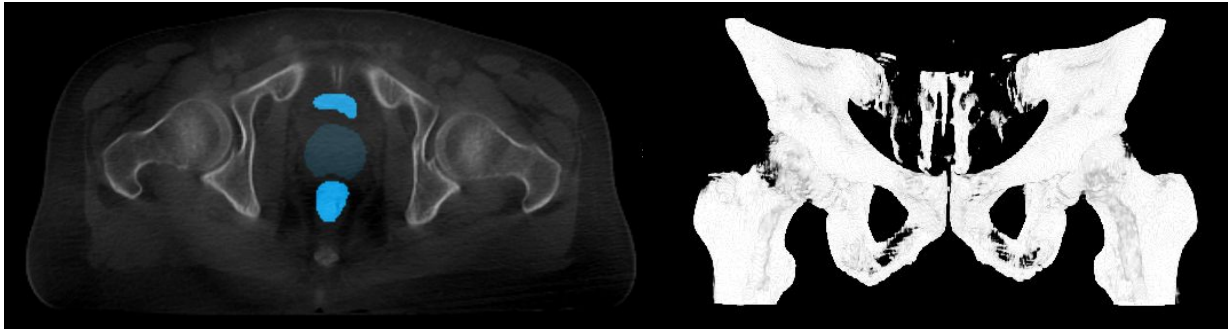
- **Task:** Import of medical data in DICOM format into visual analytics application for radiotherapy:
 - organ segmentations (contours to volumes)
 - radiotherapy plan
 - CTs
- **Technology:** Java,
<https://medevel.com/dicom-frameworks-libraries/>
- **Deliverables:** Import module + GUI



Contact: Katka

Visualization of differences in soft tissue vs. bony registration

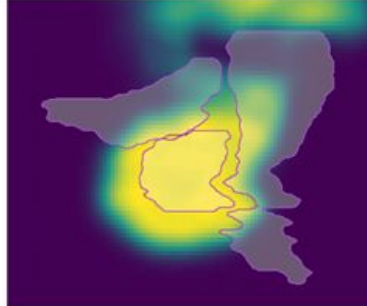
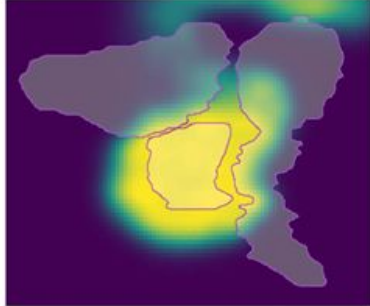
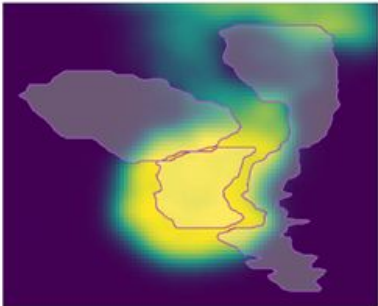
- **Task:** Visualization of differences in detected positional changes when using different rigid registration methods
 - Data preprocessing (bone segmentation, e.g., in MeVisLab)
 - Visualization of registration method impact on treatment planning
- **Technology:** Java



Contact: Katka

Deformable organ registration

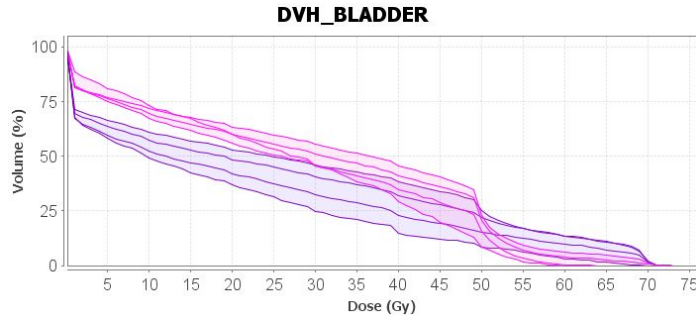
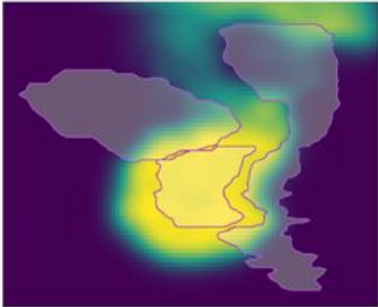
- **Task:** Implement a registration method for detection of patient organ deformations in time
 - Generation of control points on organ surfaces
 - Registration
 - Motion visualization
- **Technology:** Java, possibly OpenCV



Contact: Katka

Visualization for radiotherapy patient education

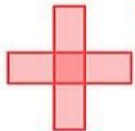
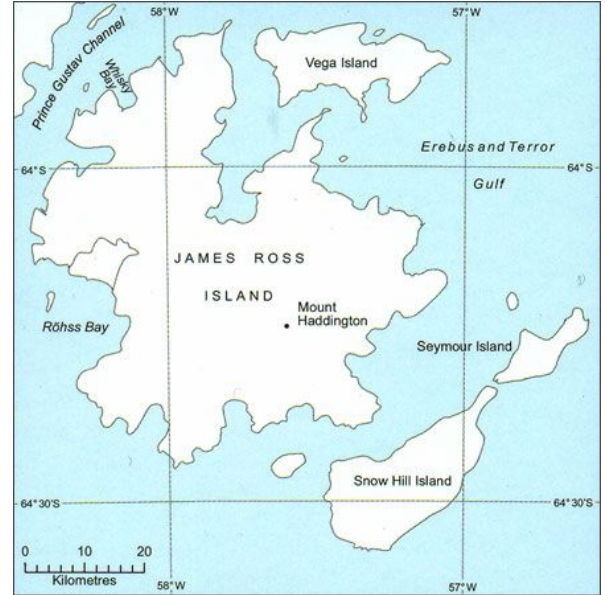
- **Task:** Design and implement methods for communication with and education of radiotherapy patients
- **Technology:** Up to you



Contact: Katka

Map visualization of measurement sites in Antarctica

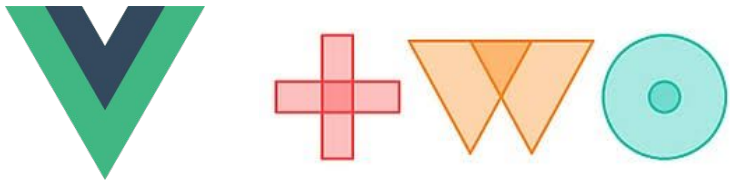
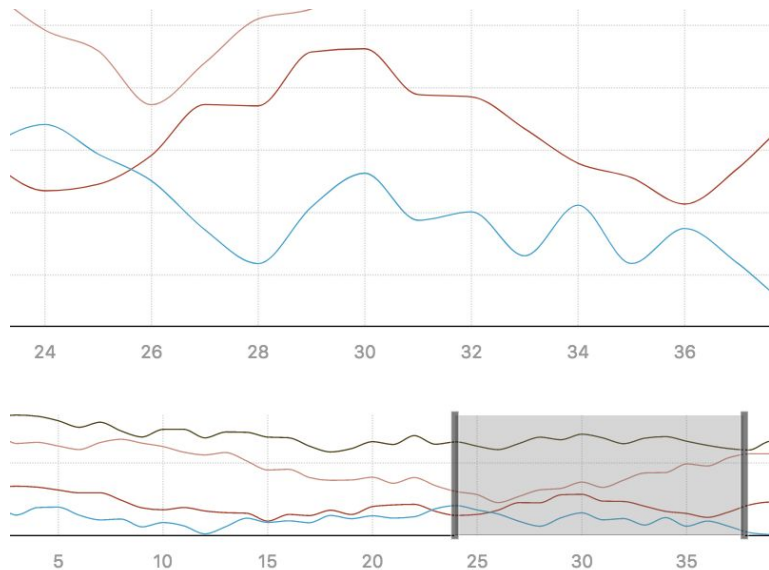
- **Task:** Create map visualization of snow height measurement sites at Mendel Polar Station in Antarctica
- **Technology:** Vue.js, Two.js, D3.js
- **Deliverables:**
 - Vue.js component



Contact: Matěj

Minimap for long charts

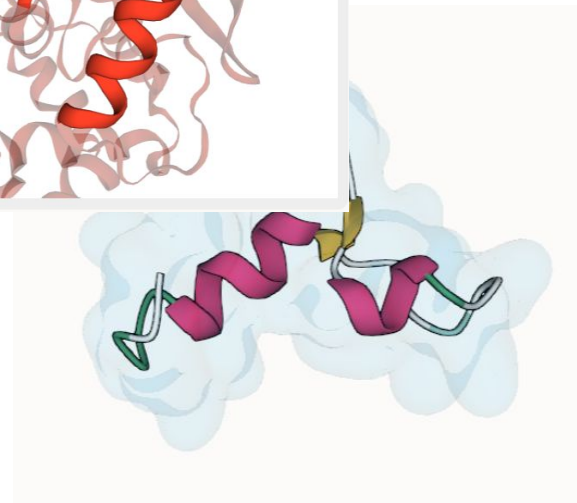
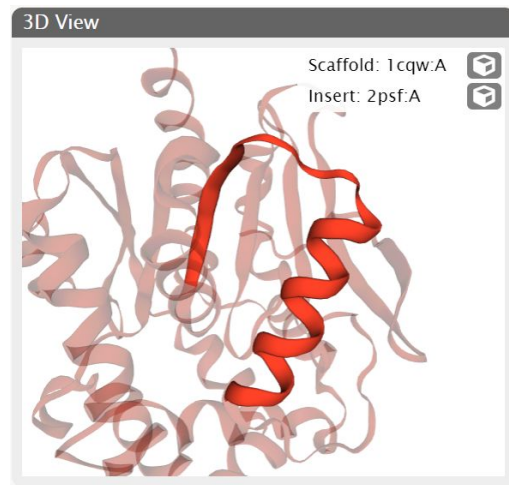
- **Task:** Create minimap for time series charts, showing current view
- **Technology:** Vue.js, Two.js
- **Deliverables:**
 - Vue.js component



Contact: Matěj

3D view in the Loop Grafter

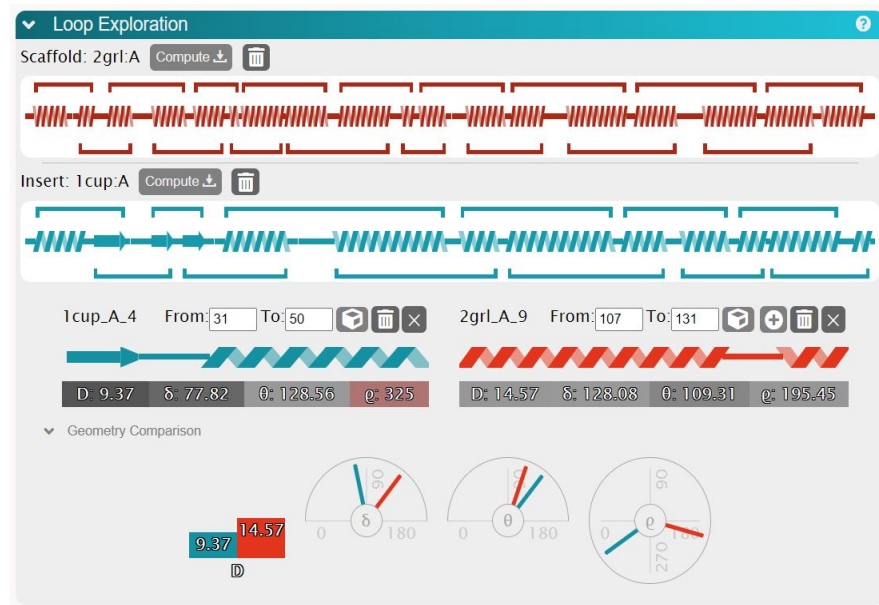
- **Background:** Interactive visual support for novel protein design method
- **Task:** The current protein 3D visualization is lacking. Refactor current implementation to use the Mol* library and implement basic interactions



Contact: Kiraa

Loop Grafter 2: The Loop Graftest

- **Background:** Interactive visual support for novel protein design method
- **Task:** Adapt the existing Loop Grafter tool for alternative workflow
- Suitable as **Bc/Mgr thesis**



Contact: Kiraa

Contacts:

- David = David Kuťák, kutak@mail.muni.cz
- Jan = Jan Byška, byska@mail.muni.cz
- Jirka = Jiří Chmelík, jchmelik@mail.muni.cz
- Palko = Pavol Ulbrich, palko@mail.muni.cz
- Simone = Simone Kriglstein, kriglstein@mail.muni.cz
- Tomáš = Tomáš Pšorn, psorn@mail.muni.cz
- Vojta = Vojtěch Brůža, bruza@mail.muni.cz
- Kiraa = Filip Opálený, kiraa@mail.muni.cz
- Katka = Katarína Furmanová furmanova@mail.muni.cz