

Slovak University of Technology in Bratislava
Faculty of Informatics and Information
Ilkovičova 2, 842 16 Bratislava 4

Team project

Engineering work

Team number: 12

Team name: 3DSpaceGen

Members: Ján Antal, Marek Drahoš, Matej Mikuš, Dominik Mazák, Peter Pápay, Andrej Písarčík, Adam Poperník

Team leader: Ing. Karol Rástočný, PhD.

Ac. year: 2018/19

Content

- 1 Big picture 2
 - 1.1 Global project goals..... 2
 - 1.1.1 Winter term..... 2
 - 1.1.2 Summer term 2
 - 1.2 Overall system architecture 2
- 2 Modules 3
 - 2.1 Virtual reality module 3
 - 2.2 Importer 6
 - 2.3 Rest API Service 7
 - Model pre-processing 7
 - 2.4 Inference machine applications 8
 - 2.4.1 Websocket module 8
 - 2.4.2 tp_3dgan module 8
 - 2.4.3 tp_sacred module..... 9

1 Big picture

In this section is presented the project created by 3DSpaceGen team. The goals of the project are presented in this document. Document describes the whole architecture, but also goes into details of each module.

1.1 Global project goals

1.1.1 Winter term

Output of our project in winter term is an application programmed in Unity for HTC Vive device. This application will support dynamic insertion of objects into scene and also manipulation of those objects. The objects will be generated with neural networks, specifically using generative adversarial networks. Inference will work in real time after accepting a request from client through provided REST Api.

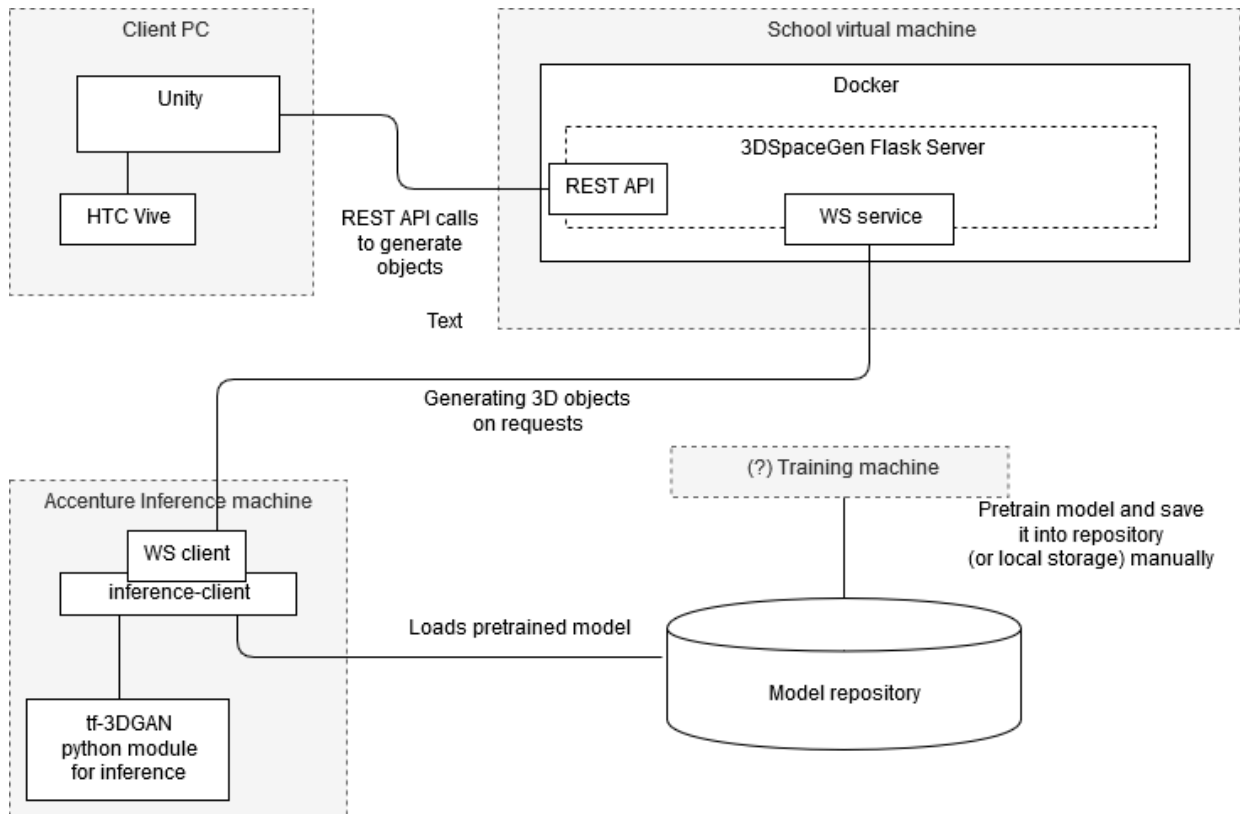
1.1.2 Summer term

The goal of summer term is to improve existing application. Main goal is to implement a feature, that user can modify the structure of generated objects. Voice control of application will also be added, to increase the user experience.

1.2 Overall system architecture

After the first five sprints we have identified, described, and implemented the following architecture. The client is a basic personal computer with Unity installed and HTC Vive connected. The user interacts with the world using the HTC Vive controllers. This client sends requests for 3D models to REST Api, which is provided on our school virtual machine. This REST Api is provided by Flask, which runs in Docker container (Flask might be replaced by Node in javascript). Flask communicates with Inference machine (Accenture PC) using the web socket, which can invoke the request for generating specific 3D model. Inference machine is used to generate 3D models from pre-trained model of GAN. This pre-trained model is loaded from shared git repository (or local storage). An agent will watch the repository for changes. When the change happens, agent updates the model on inference machine. New models will be inserted into repository by hand and they will be trained on external computers.

Communication between inference machine (IM) and school virtual machine (SVM) is described below. SVM represents a server role and IM represents client. The reason is that the IM is provided by Accenture company, which has strict security rules. After the connection is made, the roles "switches" and SVM becomes a client and sends requests to IM.



Img. 1: System architecture. The training machine is denoted with question mark, because the training machine might change in the future.

2 Modules

Repositories

All classes, libraries, files and documents which are used in our modules are in git repositories

Unity – This repository is for Virtual reality module, which includes the Importer module too.

2.1 Virtual reality module

The module allows the user to manipulate with 3D objects, change their position, size, rotation, also allows the model to be stored in the space according to the user's requirements. Through this module, the user can insert individual 3D objects into the scene.

Requirements:

- Allow the ability to select the category of the object(s) which will be generated in the scene
- Allow the ability to scaling the generated object(s)
- Allow the ability to rotate the generated object(s)
- Allow the ability to moving the generated object(s)

User stories

- As a user, I want regenerate 3D model in a middle of scene, because I want to see a new object.
- As a user, I want to move an object in my scene, because I want to customize the scene.
- As a user, I want to scale an object in my scene, because I want to customize the scene.
- As a user, I want to rotate an object in my scene, because I want to customize the scene.

Analysis

For the space visualisation, we decided that the virtual reality will be more suitable for our solution unlike the augmented reality, because it allows to change the whole space and the user is not limited by the real objects in the scene.

Design

We suggest creating a simple menu to allow the user to select the category of the desired object which will be spawned in the scene.

When the object is spawned, the user will be able to manipulate the object (change the scale, position and rotation) via the HTC Vive controllers and delete it.

Parts:

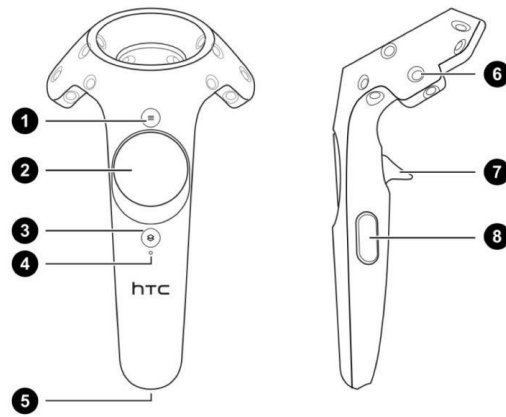
- **Unity 3D** is a multi-platform gaming engine developed by Unity Technologies. Unity provides development of 2D and 3D games also in virtual reality using SteamVR plugin. In addition to the graphical creation environment, it also supports the creation of scripts primarily in C #.
- **HTC Vive** is a virtual reality head-mounted display. It uses "room scale" tracking technology to allow the user to move in 3D virtual space and motion-tracked handheld controllers to interact with the environment.
- **VRTK** (Virtual Reality Toolkit) is a collection of useful scripts and concepts for building VR solutions rapidly and easily in Unity3D.

We use implemented VRTK solutions such as:

- Head-set menu with 2D buttons.
- Interactions like touching, grabbing and using objects
- Interacting with Unity3d UI elements through pointers or touch.

Manipulations with objects via controllers

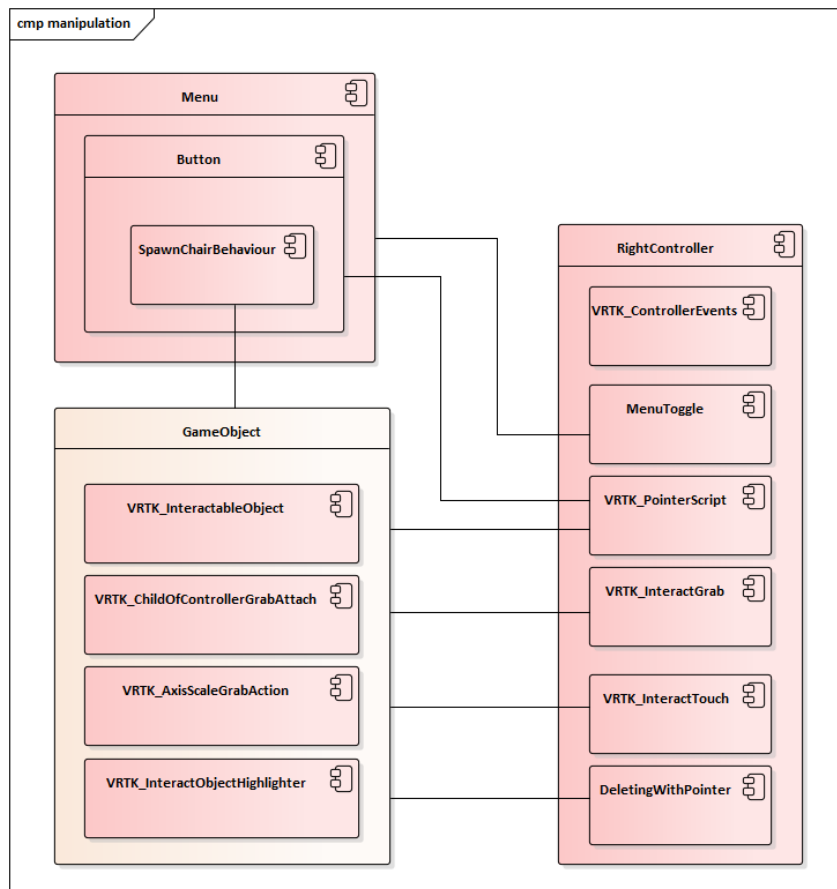
The controller has several buttons that allow user to manipulate with the application in virtual reality. The user can manipulate/interact with objects via this controlleres which are described in picture.



1	Menu button
2	Trackpad
3	System button
4	Status light
5	USB charging adapter
6	Tracking sensor
7	Trigger
8	Grip button

Img 2: HTC controllers (<http://notes.caseorganic.com/2016/11/02/htc-vive-controllers/>)

Implementation



Img 3. VR module model

Dynamics

The `VRTK_ControllersEvents` component detects all of the HTC Vive controllers buttons actions.

1. Turning the VR laser pointer

While the button 2 on the controller is pressed, `VRTK_PointerScript` calculates the collision of the ray from the controller with the other objects in the scene. This ray is rendered by another component.

2. Showing the menu

When button 1 on the HTC Vive controller was pressed, the `MenuToggle` component shows the menu with the buttons in the scene. To hide the menu, the button 1 needs to be pressed again. When the menu category is selected the importer will take care to spawn the object in the scene.

3. Spawning the GameObject

The category of the object can be selected by the VR laser pointer and confirmed by the trigger button (button 7). The selected object is generated and spawned in the scene by the `Importer` module.

4. Scaling, rotation and moving with objects

The user can grab the object with the controllers to change its position, scale, and rotation. To grab the `GameObject` the user needs to hold button 8. The `GameObject` is then assigned to the controller by the `VRTK_ChildOfControllerGrabAttach` component and follows its position and rotation. Scaling action is provided by the `VRTK_AxisScaleGrabAction`. The user needs to grab the `GameObject` with both of the controllers and move them further/closer to increase/decrease the scale. The collision between the controller and the `GameObject` is calculated by the `VRTK_InteractTouch`.

5. Deleting the GameObject

While the user is pointing on the `GameObject` with the laser pointer, the `GameObject` is loaded into `DeletingWithPointer` component. When the trigger button (button 7) is pressed the `GameObject` is deleted.

2.2 Importer

Requirements:

- This module must be able to send requests to our implemented REST API.
- This module must be able to handle the response from our REST API.
- This module must be able to create a `GameObject` from a wavefront object file received from the server as a response to a REST API request.

User story:

As a user, I want to have a service that I can trigger in order to generate a 3D model, which I can place in the scene.

Analysis

This module is responsible for importing new 3D models which are generated at runtime by the end user. into Unity game engine. These 3D objects are also placed into the scene as `GameObject` so the user can interact with them. Being able to create and import new 3D models at runtime is essential for our project. We want to allow the user to create new objects while he's inside a VR environment. We could just bundle the final product with 3D GAN neural network, pretrained models and all other modules, but running all these modules like VR application, object generation and object processing all at once at one machine would put the client machine under huge load and consume a lot of hardware resources.

Therefore, we've decided that the best solution will be to create the object on a remote dedicated server (or servers) and provide a service, which can be called by the client machine when needed. This way the client machine doesn't have to spend resources on generating the 3D object, but a reliable and fast object importer is needed for downloading and importing the newly generated object into the VR environment.

Design

The client sends a request using our predefined REST API requesting an object from a specific category. After receiving a successful reply, the reply contains the 3D object data in a wavefront object file format, which is then processed by the object importer and transformed into a GameObject instance. The new GameObject is afterwards placed in the scene, where the user can interact with the objects.

Most of the hard work, that is generating the object using a neural network and converting it into a usable format, is being done outside of the Unity game engine and the end user equipment. With this design we aim to reduce the load on the end user equipment and achieve a smoother and overall better experience for the end user.

Implementation

The client sends a HTTP POST request to the 3DSpaceGen Flask Server containing a JSON object with the specification for the requested object. The Flask Server forwards the request to the Accenture Inference machine, which generates a 3D object using a pre-trained model based on the parameters included with the request. After the object generation phase, post-processing follows. The generated object is converted from the output .MAT file to wavefront object file (.OBJ), and a second pass of post-processing minimizes the face and vertex count. The object is then sent back to the 3DSpaceGen Flask Server, which forwards the object to the client machine, running the VR setup. The client machine imports the object file and adds it into the scene.

2.3 Rest API Service

Design

We designed a Flask server as REST service that is responsible for responding to a user request, where a user specifies kind of object to generate. The request is a message, which contains various information, such as requested object type (e.g. chair, table ...) or id. Other parameters will be specified later in the project. After the message is parsed, the tf-3Dgan module is called.

Since our server is not able to generate an object (missing graphics card), we propose to use a websocket server in order to communicate with the external machine, that has an appropriate driver to perform this task. When the request from the client comes, the server determines which type of object has to be generated and sends the request via web socket to inference server. The server is then waiting for the response from the inference machine. If the inference server returned the generated object as the response, the server sends this object to the end user, otherwise sends an HTTP error message.

Model pre-processing

Analysis

The 3D model is generated by our implementation of 3D GAN neural network, which outputs a .MAT file format. This file format is not supported by Unity. Another problem with the generated objects is that it contains a huge number of vertices, with many of them being redundant. Therefore, we have identified 2 main problems, which need to be addressed before forwarding the generated model to the client machine. These problems are:

- Incompatible object file format
- Too many redundant vertices

Design

After the 3D object is generated, this model pre-processing module is called. This module consists of 2 parts – an object format converter and an object vertex count minimizer.

First, the object format converter is initialized. This algorithm builds a wavefront object file in standard .OBJ format from the source .MAT file. The resulting object file consists of vertex coordinates and face data – a vector of three indexes indicating which vertices make up the face.

Afterwards, the object simplification algorithm is called, which iterates over the vertices of the object and removes redundant vertices. It is also essential to reconstruct the faces after the vertices have been removed, which is taken care of by this algorithm as well.

The result is a wavefront object file, which is compatible with Unity game engine, with optimized vertex count.

2.4 Inference machine applications

Inference applications runs on machine provided by accenture. It consists of three modules. A **web socket module** which communicates with server and **tp_3dgan module** which implements neural networks and provides interface for other modules to interact with them, and **tp_sacred module** which runs independently of web socket and provides a training environment for the neural networks.

2.4.1 Websocket module

Repository: inference-client

Main module on the inference machine is a simple web socket client implemented in python. On the startup, this WS client will try to establish a connection with our specified school server (IP address and port are given in config file). After the connection is established, the ‘client’ will listen to requests made from ‘server’ in an endless loop. The specifics of request message are described in sections before.

On the request message arrival of type ‘generate’, tp_3dgan module is called for inference. If tf-3DGAN module produces error (not working gpu, model not found error, import error, ..), random noise object is returned to server instead.

The module consists of single **client.py** file/script. The installation is described in **README.md**. Setup of server IP address and port is necessary in top of the script.

2.4.2 tp_3dgan module

Repository: tp_3dgan

The tf-3Dgan module is a neural network implemented in tensorflow. On function call, correct pre-trained neural network model is loaded into memory from local storage (this will be optimised later in the project). Then the inference (feedforward phase in neural network of random noise) is made. Results (generated 3D object) are returned in a function.

This module consists of one single main class **models/GAN**. This class provides an interface for training, loading of model and inference for clients. Also another mini-module is used for helper functions in **utils/dataIO** for input-output operations.

This whole module is packaged in such way, that it can be easily installed using standart package manager **pip** and be used in external applications.

The module provides the following interface using standard python functions:

- **__init__(obj_type, ..)** - Constructor with optional arguments as hyperparameters. Obj_type is required and means an object type (such as chair, table...).
- **load_model(model_path)** - Loads a trained model from a specified *model_path*.
- **train(data_path, ..)** - Trains a model, with given optional hyperparameters. *Data_path* is required argument a specifies path to training data.
- **inference(..)** - Generates some examples from loaded model. Additional optional parameters may be used to vary the process of generating.

The architecture of the generative adversarial neural network is implemented as specified in paper by Wu et al. From NIPS 2016 conference. And the implementation is inspired by open source implementation from repository <https://github.com/meetshah1995/tf-3dgan>.

Reference paper:

WU, Jiajun, et al. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In: *Advances in Neural Information Processing Systems*. 2016. p. 82-90.

2.4.3 tp_sacred module

Repository: tp_sacred

Last application used on inference machine is tp_sacred. This application is responsible for the whole pipeline of training, deploying and storing the trained models. It uses the **sacred** framework to isolate a single training run. Main script loads a configuration file of training hyperparameters and then starts the training.

Training process initializes a **GAN** object from **tp_3dgan** module. Then it loads it with the provided dataset. For training, we have used 3DShapeNets dataset of objects, which can be freely downloaded from internet. The dataset consists of thousands of simple 3D objects (e.g. chairs, tables, toilets) in various angles.

When the training is finished, the trained tensorflow model is saved into local directory and an agent sends a slack notification to team slack channel about training finish. Also couple of objects are generated and posted to our hosted **visdom** server for visualisation. The **visdom** server is a simple docker container running on our school virtual machine. Visdom is a framework developed by facebook in python for visualisation of various graphs and models.

Tímový projekt

Riadenie projektu

Číslo tímu: 12

Názov tímu: 3DSpaceGen

Členovia: Ján Antal, Marek Drahoš, Matej Mikuš, Dominik Mazák, Peter Pápay, Andrej Písarčík, Adam Poperník

Vedúci tímu: Ing. Karol Rástočný, PhD.

Ak. rok: 2018/19

Obsah

1. Riadenie projektu	2
1.1 Big Picture	2
1.1.1 Role členov	2
1.1.2 Aplikácie manažmentov	3
1.1.3 Sumarizácia šprintov	3
1.2 Globálna retrospektíva	4
1.3 Motivačný dokument	5
1.3.1 Tím	5
1.3.2 Motivácia	6
1.3.3 Zoradenie tém podľa priority	7
1.3.4 Rozvrh tímu	8
1.4 Metodiky	8
1.4.1 Metodika komunikácie	8
1.4.2 Metodika manažmentu tímu	8
1.4.3 Metodika verziovania projektu	8
1.4.4 Metodika písania zdrojového kódu	8
1.5 Webové sídlo	8
Príloha A: Export úloh	9
Export úloh prvého šprintu	9
Export úloh druhého šprintu	9
Export úloh tretieho šprintu	10
Export úloh štvrtého šprintu	11
Export úloh piateho šprintu	13

1. Riadenie projektu

1.1 Big Picture

Dokument vznikol ako výsledok práce tímového projektu 3DSpaceGen po prvom semestri, za účelom zdokumentovania celkovej práce na projekte. Primárnym cieľom tohto dokumentu je opísať manažment tímu, detailnú špecifikáciu softvérového produktu bližšie popisuje dokument inžinierske dielo.

Tento dokument popisuje podrobný priebeh riadenia tímového projektu 3DSpaceGen po prvých troch šprintoch s využitím metodiky Scrum. V dokumente sa nachádzajú podrobne popísané manažérske role jednotlivých členov tímu, vzhľadom na ich úlohy. Dokument ďalej obsahuje popis šprintov, podiel práce každého z členov tímu, metodiky práce a záverečný pohľad na jednotlivé retrospektívy. Súčasťou dokumentu je aj export úloh získaný z nástroja TFS, ktorý bol použitý pre jednoduché a rýchle riadenie tímu.

1.1.1 Role členov

Tím tvorí 7 členov, product owner Vladimír Hlaváček a vedúci tímu Karol Rástočný. Pre manažment tímu používame nástroj Microsoft Team Foundation Server (TFS). S týmto nástrojom interaguje každý člen tímu a každý má aj isté zodpovednosti. Špecifikácia metodiky práce je definovaná v sekcii Metodiky tohto dokumentu.

Adam Poperník: Frontend programmer, C# programer, UI/UX designer, webpage admin
Adam je zodpovedný za webovú stránku projektu. Okrem toho sa podieľa na vývoji VR aplikácie.
V dokumentácii vypracoval: Implementácia

Ján Antal: Manažment UI/UX, C# programmer, VR developer
Zodpovedný za návrh používateľského rozhrania v Unity. Ján sa hlavne venuje primárne vývoju VR aplikácie v HTC Vive a pracuje najmä s Unity.
V dokumentácii vypracoval: Analýza

Andrej Pisarčík: Manažér komunikácie
Zodpovedný za manažment komunikácie. Definoval pravidla komunikácie v Slacku.
Andrej sa venuje testovaniu existujúcich riešení a open source programov na testovanie GANiek. Okrem toho sa venuje aj implementácií Flask serveru v pythone.
V dokumentácii vypracoval: Úvod, Návrh

Dominik Mazák: Manažment konfigurácií systému
Dominik je zodpovedný za administráciu školského serveru a Docker containerov. Zároveň je zodpovedný aj za DevOps ako CI (Continuous Integration ...).
V dokumentácii vypracoval: Aplikácie manažmentov, Globálna retrospektíva

Matej Mikuš: Manažér výskumu
Matej je hlavný researcher tímu a má zodpovednosť za výskum 3D GAN. Venuje sa vyhľadávaniu nových riešení a rôznych open source programov v oblasti GAN sietí. Okrem toho sa podieľa pri vývoji Flask serveru.
V dokumentácii vypracoval: Testovanie, Motivačný dokument

Peter Pápay: SCRUM Master, python programmer
Zodpovedný za manažment riadenia práce v tíme. V tímovom projekte definoval pravidlá práce s TFS.

Peter je SCRUM master a je zodpovedný za celkovú prácu s nástrojom pre manažment tímu. Okrem toho programuje v pythone a venuje sa implementácií GAN siete.

V dokumentácii vypracoval: Role členov, Sumarizácia Šprintov, Celkový pohľad na systém

Marek Drahoš: Manažér verzii

Zodpovedný za manažment verzii.

Marek je backend programátor a je zodpovedný za tok dát v projekte.

V dokumentácii vypracoval: Úvod, Globálne ciele projektu, Export úloh a Review

1.1.2 Aplikácie manažmentov

Slack

Slack je sada kolaboračných služieb a nástrojov uložených na cloude. Medzi hlavné funkcie Slacku patria:

- Perzistentné chatové miestnosti
- Súkromné chatové miestnosti
- Súkromné správy
- Vytváranie tímov
- Zdieľanie súborov
- Vyhľadávanie

V rámci nášho projektu je Slack našim hlavným spôsobom komunikácie a organizácie v rámci tímu. Využili sme funkciu vytvárania tímov, ktorú Slack poskytuje, a vytvorili sme si vlastný tím, kde sa nachádzajú všetci členovia tímu vrátane vedúceho projektu a product owner-a. Ďalej Slack používame aj na príjem notifikácií. Tak isto máme vytvorených viacero špecifických komunikačných kanálov určených pre užšiu podskupinu členov tímu, ktorí pracujú na rovnakej časti projektu. V týchto špecifických kanáloch sa členovia Tieto kanály sú bližšie popísané v metodike komunikácie.

Team Foundation Server

Team Foundation Server (ďalej už len TFS) je produkt od spoločnosti Microsoft, ktorý poskytuje služby ako manažovanie zdrojového kódu, verziovanie, reporting, automatické buildovanie a deployment, manažment projektu. Celkovo je TFS našim základným prvkom agilného vývoja softvéru. Činnosti, ktoré v TFS vykonávame v rámci nášho projektu sú vytváranie user stories, kde zadefinujeme určitú časť funkcionality alebo určitý cieľ, ktorý chceme dosiahnuť. Ďalej si rozbijeme tieto user stories na jednotlivé úlohy, ktoré sa priradia členom tímu. Následne si vytvoríme plán šprintu. Na základe ohodnotenia zložitosti jednotlivých user stories a tieto si následne pridelieme do šprintu. Toto nám umožňuje efektívne plánovať jednotlivé šprinty a sledovať priebeh práce jednotlivých členov tímu na svojich úlohách.

Na serveri TFS sme si založili našu vlastnú Wiki, ktorú počas práce na projekte neustále rozširujeme. Služi nám ako jednotné miesto pre zbieranie všetkých relevantných znalostí k téme projektu. TFS sme taktiež použili aj na vytvorenie automatického releasovania softvéru na cieľové platformy.

K nástroju TFS sme napísali aj metodiku, ktorá sa nachádza tu.

Google kalendár

Nástroj google kalendár používame na plánovanie. Náš tím má založený jednotný kalendár, v ktorom si vyhradíme presný čas určený na prácu na jednotlivých úlohách

1.1.3 Sumarizácia šprintov

V projekte sme absolvovali troje dvojtýždňové šprinty. Po prvom šprinte sme v retrospektívne identifikovali malé problémy v tíme (spôsob komunikácie, práca s TFS), ktoré sa nám v druhom šprinte

podarilo opraviť. Zároveň sme opakovane objavovali problémy s priebežnou prácou a seba organizáciou. Tieto problémy plánujeme vyriešiť v letnom semestri.

Šprint 1 - Londýn

Prvotný šprint v ktorom sme si stanovili naše ciele, premysleli možné riešenia a pripravili si vývojové prostredie.

Šprint 2 - Paríž

Vytvorenie VR aplikácie v Unity s možnosťou pridania objektov. Príprava flask serveru a REST Api. Študovanie rôznych vedeckých prác na spracovanie 3D objektov pomocou GAN sietí.

Šprint 3 - Berlín

Príprava architektúry celkového systému. Manipulácia s objektami vo VR (premiestňovanie, rotovanie, zmena veľkosti). Testovanie rôznych ďalších GAN sietí a kontrola ich výsledkov.

Šprint 4 - Bratislava

Šprint v ktorom sme vo VR vytvorili ovládanie objektov - scalovanie, delete, presun objektu. Rozbehanie websocketu pre komunikáciu s inferenčným strojom

Šprint 5 - Štokholm

Implementácia celkovej pipeline od klienta až po inferenčný stroj. Setup CI na kľúčové repozitáre v projekte.

1.2 Globálna retrospektíva

V zimnom semestri sme identifikovali niekoľko problém v rámci organizácie. Tieto problémy sa niekoľko krát vyskytli aj v retrospektívach.

Čo sa nám páčilo:

- Práca s technológiami (virtuálna realita, neurónové siete)
- Práca na zaujímavých úlohách
- Po upozorneniach sa zapracovalo na lepšej tímovej komunikácii (stand-up, lepšia koordinácia)
- Upozornenia prostredníctvom Slacku - lepší prehľad o vykonávanej práci a nasadzovaní (kto kedy commitoval, pridal úlohu, zmenil jej stav a pod.)
- Práca bola oveľa jednoduchšia ak boli definované jasné požiadavky - študent vedel čo sa od neho očakáva, čo malo za následok vyššiu časovú efektívnosť

Čo sa nám nepáčilo:

- Nedostatočne výkonný stroj na prácu/testovanie neurónových sietí
- Neskoré riešenie úloh
- Nízka výpovedná hodnota informácií k taskom, user story (občas sa nevedelo, čo sa od nás očakáva pri riešení daného problému)
- Často krát zlá komunikácia medzi jednotlivými členmi tímu, vzájomné neporozumenie
- Neefektívnosť, ktorá vznikala stretávaním všetkých členov aj keď boli vytvorené skupinky, ktoré dokázali pracovať na svojej časti aj osobitne
- Ako študenti sme mali častokrát veľmi málo času, kedy sa bolo možné skutočne venovať práci (veľa zadaní aj z iných predmetov, žiaden voľný čas, ktorý by mohol byť venovaný tímovému projektu)
- Súčasná práca viacerých členov na jednej úlohe - neefektívnosť

- Málo commitov k jednotlivým modulom a programovým častiam (nízka potreba využívania Git-u)
- Zlý burndown chart - nevyváženosť práce na úlohách (práca na úlohách iba počas jedného dňa v týždni)
- Ťažké estimovanie user story ale aj jednotlivých úloh

Čo by sme mali zlepšiť:

1. Nevykazovanie časov / neprideľovanie taskov
 - Vyriešené: dohoda, že čas sa vykazuje najneskôr v daný deň
2. Dochvilnosť - neskoré príchody na tímové stretnutia
 - Riešenie: Zavedenie pokladničky
3. Priebežná práca, stand-up
 - Riešenie: povinný stand-up, pokladnička
4. Málo efektívna práca na stretnutiach
 - Riešenie: rozdelenie do menších skupín, 2 hodiny spolu + 2 oddelene
5. Nepravidelnosť code review
 - Riešenie: nastavenie schvaľovania pull requestov
6. Nesplnenie definition of done na uzavretých taskoch/user story
 - Riešenie: prísnejšie dodržiavanie pravidiel

1.3 Motivačný dokument

Kontaktný mail: timovyprojekt@centrum.sk

1.3.1 Tím

Náš tím pozostáva z členov: Ján Antal, Marek Drahoš, Dominik Mazák, Matej Mikuš, Peter Pápay, Andrej Pisarčík a Adam Poperník.

Za veľké pozitívum považujeme vzájomné spoluprácu členov nášho tímu, kedy sme mali možnosť priamo počas štúdia pracovať na viacerých projektoch. Na druhej strane udržiavame spoločné kontakty aj mimo školy, čo vo veľkej miere upevňuje súdržnosť nášho tímu.

Väčšina členov nášho tímu prišla počas bakalárskeho štúdia do kontaktu s oblasťou strojového učenia, či už išlo o samotnú bakalársku prácu alebo predmet inteligentná analýza údajov. Konkrétne sme v projektoch využívali neuronové siete za pomoci frameworkov, ako napríklad - CNTK (Matej), Keras (Marek), TensorFlow (Peter). Vieme pracovať s relačnými (MySQL, PostgreSQL) a nerelačnými (MongoDB, Firebase) databázami. Okrem toho máme skúsenosti so správou sietí (Dominik), spracovaním obrazu pomocou knižnice OpenCV (Ján) a SimpleITK (Matej), paralelným či aynchrónnym programovaním, JEE klient-server architektúrou a programovaním elektronických zariadení s využitím hardvéru Arduino (Adam).

Medzi predmety, ktoré by nám mohli pomôcť k úspešnému vypracovaniu tímového projektu patria objavovanie znalostí, pokročilé databázové technológie, vyhľadávanie informácií, neuronové siete, počítačové videnie, distribúcia obsahu v internete, štatistické metódy vyhodnocovania experimentov.

V rámci mimoškolskej aktivity nám bolo umožnené nadobudnúť dôležité skúsenosti aj v praxi. Predovšetkým v oblasti vývoja webových aplikácií, kde sme mali možnosť vyvíjať ako frontend (Angular, React, HTML, CSS, Javascript, JQuery, Bootstrap), tak aj backend (Java Spring, PHP, Python Flask, ASP.NET Core, Node.js). Pracovali sme agilným vývojom a princípy SCRUM-u nám vôbec nie sú cudzie (Marek, Matej, Andrej, Peter, Adam). Matej a Peter sa podieľali na vývoji interaktívnych chatbotov integrovaných do aplikácii Slack a Skype. Ovládame prácu s verziovacím systémom Git, systémom pre deployment Docker (Andrej a Peter) a Linux serverom (Andrej). Navyše máme skúsenosť

s project management systémami JIRA (Andrej, Adam), VSTS (Matej). Jeden člen nášho tímu riešil taktiež štátne zákazky, napríklad pre Ministerstvo dopravy, kde sa podieľal na vytváraní informačného systému pre cestnú dopravu (Adam).

Vo voľnom čase sa radi venujeme aktuálnym technológiami, ktoré nám umožňujú rozširovať si svoje obzory. Ako tím máme radi výzvy, s čím priamo súvisí naše nadšenie pre objavovanie a prehlbovanie znalostí.

1.3.2 Motivácia

Monitoring antisociálneho správania [MonAnt]

Sociálne siete sú súčasťou dnešného každodenného života. Niet pochýb, že sú veľkým pomocníkom, no podobne ako iné médiá aj sociálne siete sú zneužívané v prospech tretej osoby. Príkladom je šírenie dezinformácií, hoaxov alebo “fake news”. Dôvodom na ich publikáciu je zvädzanie čitateľa s falošnými informáciami a teda neférovou ovplyvňovať verejnú mienku a snažiť sa pretlačiť svoju agendu. Takto sa to stalo aj v roku 2016 pri prezidentských voľbách v US. Veľké spoločnosti ako Facebook a Google boli prvotne kritizované za nečinnosť, no obe začali vynakladať obrovské peniaze na boj proti fake news.

Myslíme si, že šírenie fake news je veľkým problémom demokratických krajín, vrátane Slovenska. Príkladom môžu byť aj často nepodložené výroky politikov, ktoré sa dostávajú do médií a rozdeľujú našu spoločnosť. Články nemožno len tak cenzurovať kvôli slobody tlače. Čo však môžeme urobiť je upozorniť na to, že nie všetko v nich môže byť pravda. Prvým krokom je detekcia takýchto článkov. Ale tým to nekončí, rovnako dôležitou súčasťou je aj ich prezentovanie verejnosti čo možno najjednoduchšou formou.

Túto tému sme si zvolili nielen z dôvodu, že si vážime a chceme zachovávať demokratické hodnoty na Slovensku, ale zároveň by sme chceli pomocou detekcie a upozorňovania na zavádzajúce a nepravdivé informácie aj pozdvihnúť úroveň mediálneho obsahu na internete. Veríme, že prácou na tomto projekte môžeme prispieť k zníženiu spoločenského napätia ktoré tu v súčasnosti panuje.

Z technického hľadiska máme v tíme niekoľko ľudí ktorí sa venujú front-endu a majú skvelé nápady ako realizovať front-endovú časť projektu. Zároveň je projekt zaujímavý aj z back-end časti, kde je potrebné crawlovať web a budovať dataset. Táto časť zaujala práve tých z nás, ktorí sa venujú umelej inteligencii a veľkým dátam.

Vyhľadávanie pomocou obrázkov [ImageSearch]

Častokrát sa stane, že si vo voľnom čase niečo odfoťíme, či už v prírode, v obchode, skrátka všade. V dnešnej dobe má už skoro každý človek fotoaparát alebo mobil, pomocou ktorého vie zaznamenať obraz a uložiť ho na svoje úložisko, prípadne do databázy. Obraz podobne ako aj text zobrazuje (poskytuje) človekovi informáciu. Niekedy sa však stane, že človek nemá dostatok znalostí o danom objekte - predstavme si hríb v lese - nevieme ako sa volá, no chceme zistiť či je jedlý. Keby sme teda vedeli jeho názov, zadáme túto informáciu do Google alebo iného vyhľadávacieho nástroja. Keďže ale nevieme slovne identifikovať daný objekt, môžeme využiť práve vyhľadávanie pomocou obrazu/obrázkov.

Niekedy však nechceme predmet iba vyhľadať a zistiť čo to je. Vyhľadávanie podľa obrázka nám už v súčasnosti poskytuje napríklad Google. Určite sa už každý z nás aspoň raz ocitol v situácii, keď narazil na nejaký zaujímavý predmet a povedal si “toto by som chcel mať!”. A tu prichádza na rad otázka, kde ten predmet kúpiť. Vyhľadávanie v e-shopoch v drvivej väčšine zahŕňa vyhľadávanie na základe textu alebo kľúčových parametrov. Preto sme sa ako tím rozhodli, že by sme chceli toto vyhľadávanie vylepšiť a využiť vyhľadávanie pomocou obrázkov na vytvorenie systému schopného tento spôsob vyhľadávania aplikovať na tovar v e-shopoch. Vidíme v tom skvelú príležitosť, ako

vytvoriť niečo zaujímavé a zároveň praktické do života. Človek by si mohol čokoľvek čo sa mu zapáči jednoducho odfoťiť a behom pár sekúnd zistiť, v ktorom e-shope to ponúkajú.

V našom tíme sa už viacerí zaoberali prácou s obrazom, najmä počas bakalárskych prác (identifikácia tumoru z obrazu) ale aj počas predmetov, ktoré sme počas štúdia absolvovali, či plánujeme absolvovať (napr. počítačové videnie, princípy počítačovej grafiky a spracovania obrazu). Aj to bol jeden z dôvodov, prečo sme sa rozhodli pre danú tému. Práca s obrazom má určite v budúcnosti veľkú perspektívu a dá sa na ňu nahliadať z viacerých aspektov.

1.3.3 Zoradenie tém podľa priority

Po prekonzultovaní sme zmenili priority tém. Našou najväčšou prioritou je projekt **MonAnt**, ktorý sme zvolili aj v systéme YonBan.

Odovzdané poradie v dotazníku:

1. Importer
2. MonAnt
3. ImageSearch
4. SmartMobility
5. TxtEnv
6. GridBox
7. In-memory-DB
8. Breyslet
9. Behamatrix
10. SK-CZ-TEXT
11. VReduction
12. FIIT-DU

Finálne poradie priorít tém:

1. MonAnt
2. ImageSearch
3. Importer
4. SmartMobility
5. TxtEnv
6. GridBox
7. In-memory-DB
8. Breyslet
9. Behamatrix
10. SK-CZ-TEXT
11. VReduction
12. FIIT-DU
13. Holographic Eyes
14. WFuntoro
15. 3DSpaceGen
16. VizReal
17. 3D-UML
18. BioMonitor
19. IoTesting
20. 3D futbal

1.3.4 Rozvrh tímu

<https://docs.google.com/spreadsheets/d/1ZCgnhet00VVJn3e0Jv9qCxUQdWshURk8J8V6ZgAk8o/edit#gid=0>

1.4 Metodiky

1.4.1 Metodika komunikácie

Táto metodika určuje pravidlá pre komunikáciu členov tímu medzi sebou, ako aj s vedúcim. Komunikáciu prebieha prostredníctvom komunikačného nástroja Slack. Pravidlá metodiky sú definované v dokumente Metodika Komunikácie, ktorý sa nachádza na webovej stránke tímu.

1.4.2 Metodika manažmentu tímu

Manažment tímu zahŕňa najmä nastavenie celého systému riadenia, procesov a výkonnosti tímu. V manažmente tímu hrá hlavnú rolu Scrum master, ktorý je zodpovedný za plynulý priebeh tímových stretnutí, plánovanie a taktiež dohliada na retrospektívu. Okrem týchto povinností sa snaží pomôcť tímu s vyriešením vzniknutých problémov počas šprintu. Scrum master zodpovedá taktiež za dodržiavanie vytvoreného plánu, pričom môže napomenúť členov tímu, ktorí nespĺňajú dohodnutý plán. Všetky pravidlá metodiky sú definované v dokumente Metodika manažmentu tímu, ktorý sa nachádza na webovej stránke tímu.

1.4.3 Metodika verziovania projektu

Metodika určuje pravidlá práce s verziovacím nástrojom Git, ktorý je určený na správu verzií zdrojových kódov. Metodika obsahuje všeobecné pravidlá, ktoré je nutné dodržiavať pri práci s ktorýmkoľvek repozitárom v rámci projektu. Metodika je bližšie opísaná v dokumente Metodika Verziovania, ktorý sa nachádza na webovej stránke tímu.

1.4.4 Metodika písania zdrojového kódu

Metodika určuje pravidlá písania zdrojového kódu v jazykoch Python a C#, ktoré tím používa v projekte. Tieto pravidlá je potrebné dodržiavať, aby ostal kód čitateľný a prehľadný. Detaily sa nachádzajú v dokumente Metodika písania zdrojového kódu, ktorý je na webovej stránke tímu.

.5 Webové sídlo

Odkaz na zip súbor, obsahujúci webovú stránku tohto tímu:

<https://uloz.to!/K4CsXWhGI8WK/webpage-zip>

Príloha A: Export úloh

Export úloh prvého šprintu

ID	Title	Assigned To	State	Description
8918	Continuous deployment of the web page	Dominik Mazak	Closed	As the product owner, I would like to have the webpage automatically deployed into Apache root after commit to master branch. TODO: * Download build agent from TFS * Deploy build agent to server * Set-up build definition in TFS
8920	Build result notification - web page	Andrej Pisarcik	Closed	We want to receive automated notifications when build has finished in a separate Slack channel. TODO: * Create appropriate Slack channel. * Get channel authorization key from Slack. * Set-up notification rule in TFS with the authorization key.
8921	Task status change	Andrej Pisarcik	Closed	We want to receive automated notifications when task status changed in a separate Slack channel. TODO: * Create appropriate Slack channel. * Get channel authorization key from Slack. * Set-up notification rule in TFS with the authorization key.
8924	As a user, I want to see empty 3D space in HTC Vive because I want to fill it later	Jan Antal	Closed	TODO: * Set-up empty Unity project for HTC Vive. * Enclose the scene into a big black box. * Pair with HTC Vive. * Run.
8925	As a user, I want to see a chair in the middle of the scene because I want to have implementation ready for 3D model visualization	Adam Popernik	Closed	TODO: * Find free online 3D model repository with appropriate model file format. * Download 3D model of a chair. * Import the 3D model into Unity project. * Place the chair into the middle of the scene, scale the chair appropriately. * Run.
8927	As a developer, I want to have a manual for neural networks because I want to use them in the future development	Matej Mikus	Closed	TODO: * Set-up Wiki in TFS. * Create a Wiki page with neural networks overview. * FFNN * RNN * CNN * GAN * Create a simple neural network demonstration.
8928	As a developer, I want to replicate the implementation from 3D GAN research paper because I want to try it out	Peter Papay	Closed	TODO: * Read the paper. * Clone the repository. * Set-up the environment. * Run the experiment. * Verify results.

Export úloh druhého šprintu

ID	Title	Assigned To	State	Description
9214	As a user, I want to see progress of 3DSpaceGen team on website, because I am interested.	Adam Popernik	Closed	* názov témy * riešitelia a základné informácie o nich (napr. z ponuky) * odkazy na doteraz vypracovanú dokumentáciu vrátane záznamov zo stretnutí (záznamy zo stretnutí budú priamo čitateľné na webe, t.j. najlepšie vo formáte pdf, prípadne HTML) Remaining: * plán projektu (na semester) * aktuálny stav plnenia plánu (t.j., úlohy, ktoré vyplynuli zo stretnutí, ich plnenie a vzťah k plánu) roly jednotlivých členov tímu (aj dočasné) * všetko zaujímavé v súvislosti s projektom a postupom prác na projekte, napr. odkazy na iné zdroje súvisiace s témou projektu
9215	As a user, I want to have service to trigger generate 3D model, so I can place it in the scene.	Dominik Mazak	Closed	Setup FLASK service on OUR server. It will contain pretrained model. Model generates an object. (off to obj, command line tool exists) Object needs to be converted to correct format. Service returns the object in correct format. Risks: Converter problems
9216	As a user, I want regenerate 3D model in a middle of scene, because I want to see a new object.	Jan Antal	Closed	When button is pressed on controller, request is made to service and returns an object. The object is spawned in the scene. * learn how to interact with the VR controllers in unity. * create the menu using the game objects (2D plane with text or image) to specify the object which we want to generate. * for each menu item assign the specific

ID	Title	Assigned To	State	Description
				text (name of the object). * implement the VR pointer to see on which menu item am I pointing with the controller. * add the action to the controller button - find the position on which the controller is pointing and print the object name in the console.
9220	As a developers, we want to get a TATRA bank grant, so we can run models on perfect machine.	Adam Popernik	Closed	Write a request (2-3 pages) based on Tatra banka requirements. Try to build compatible hardware and get the prices. Deadline in 2nd of November. http://www.nadaciatatrabanky.sk/index.php/grantove-programy/e-talent/ Pridat https://www.fiit.stuba.sk/sk/fakulta-informatiky-a-informacnych-technologii/prihovor-dekana-fiit-stu/zriadovacia-listina.html?page_id=416
9221	As a team, we want to participate in TP cup, to get honor and glory in FIIT STUBA.	Peter Papay	Closed	http://www2.fiit.stuba.sk/tp-cup/ * Tím sa prihlási do súťaže vypracovaním a odovzdaním prihlášky v rozsahu max. 2 strán A4 (text fontom Times New Roman, 11pt). * Prihláška predstavuje prvotný návrh riešenia projektu * Obsahuje * kontakt na tím (minimálne mail), * stručné predstavenie tímu, * základnú motiváciu, * opis kontextu a náplne projektu, * stanovenie cieľov projektu, požiadaviek na výsledný produkt, ako aj spôsob a postup akým budú dosiahnuté. * Cieľom je presvedčiť, že: * ste identifikovali zaujímavý problém, * zamýšľané riešenie rieši tento problém, * ste tí správni pre riešenie projektu, * viete aké technológie a postupy využijete na jeho realizáciu.
9384	As a developer, I want to replicate the implementation in python3.6 from 3D GAN research paper, because I want to try it out.	Peter Papay	Closed	TODO: * Set-up the environment. * Reimplement to python 3.6 * Write wiki * Run the experiment. * Verify results. * Commit * Push * Pull request * Ask friend for review * Merge
9386	As a customer, I want to see analysis of state of the art GAN, because I want to be sure that we select best solution.	Matej Mikus	Closed	Read about recent GAN papers from github repository in wiki. Write about it to Wiki. Try to find interesting survey papers. What dataset was used, what models, parameters and short summary for every paper.
9387	As a developers, we should contact google/aws/azure to get better machines, because we want to use them.	Andrej Pisarcik	Closed	* Write the general application (can be derived from TP Cup) * We should learn the requirements for applications for every vendor * Write the application * send the mail from @ stuba.sk domain

Export úloh tretieho šprintu

ID	Title	Assigned To	State	Description
9600	As a user, I want to move an object in my scene, because I want to customize the scene.	Adam Popernik	Closed	The created object will be spawned in middle of scene The user then can grab him a move the object, and put the object down
9601	As a user, I want to rotate an object in my scene, because I want to customize the scene.	Jan Antal	Closed	User can selected object in scene User then can rotate the object User then can place it down
9602	As a user, I want to scale an object in my scene, because I want to customize the scene.	Jan Antal	Closed	The created object will be spawned in middle of scene The user then can grab him a scale the object, and put the object down
9765	As a user, I want regenerate 3D model in a middle of	Jan Antal	Closed	When button is pressed on controller, request is made to service and returns an object. The object is spawned in the

ID	Title	Assigned To	State	Description
	scene, because I want to see a new object.			scene. * learn how to interact with the VR controllers in unity. * create the menu using the game objects (2D plane with text or image) to specify the object which we want to generate. * for each menu item assign the specific text (name of the object). * implement the VR pointer to see on which menu item am I pointing with the controller. * add the action to the controller button - find the position on which the controller is pointing and print the object name in the console.
9766	As a developer, I want to have working implementation of tf-3dgan on windows machine, because I probably have no other choice.	Peter Papay	Closed	Make tf-3dgan run on windows machine. https://docs.microsoft.com/en-us/windows-hardware/drivers/display/timeout-detection-and-recovery https://stackoverflow.com/questions/17186638/moving-registry-to-increase-gpu-timeout-windows-7 https://stackoverflow.com/questions/16317505/cublas-failed-to-synchronize-stop-event
9767	As a user, I want to see implementation of other 3dgans, because I want to see the results compared.	Matej Mikus	Closed	One more implementation is used.
9768	As a team, check out external box, because we might buy one.	Dominik Mazak	Closed	* Extern GPU through USB-C / USB3/ Thunderbolt * So we can try and run it on our notebooks / labtops * Find more alternatives of boxes and GPU's
9769	As a team, we want to have methodics (git, tfs, communication) and guidelines, because we want to be more organised.	Peter Papay	Closed	* Write down methods for GIT /etc (more in Sprint 02 Retrospective)
9770	As a team, we must have sprint 1-3 documented, because we are obliged to for subject.	Marek Drahos	Closed	Inzinierske dielo https://docs.google.com/document/d/1eLUq6wszjWF BjZw 5UAGH2njD4jZa8jNh3fhg8XUvvI/edit Riadenie projektu https://docs.google.com/document/d/1khozBT5Etd C7H6UezrEjvTCNEhnAEmZuigYh-I7a8XY/edit for more information see: http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/materialy/dokumentacia2017-18.pdf deviaty týždeň semestra (podľa rozvrhu stretnutí, pracovný deň nasledujúci po dni v týždni, kedy má tím plánované stretnutia), najneskôr však 23.11.2018, 12:00 (tí čo máte stretnutia v piatok, naplánujte si to tak, aby ste to zvládli) odovzdanie dokumentácie po prvých troch šprintoch (vrátane celkového obrázku projektu) plus dokumentácia k riadeniu - elektronicky * Úvod - o čom je tento dokument, ciele, ohraničenia. * Globálne ciele projektu na zimný semester * Celkový pohľad na systém (architektúra, dátový model, diagramy tried, moduly plus referencie/zoznam priložených e-dokumentov) Retrospektiva a aj vyjadrenie ako sprint prebiehal (co sa spravilo, co sa nedstihlo, pridať nejaký burndown chart, spraviť vyjadrenie či sme spravili toho viac alebo menej, info k organizácii timoveho projektu atd) http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html#odovzd-zs

Export úloh štvrtého šprintu

ID	Title	Assigned To	State	Description
9599	As a developer, I want to have a build/train/deploy pipeline for neural networks, because	Peter Papay	Closed	https://github.com/IDSIA/sacred Setup the accenture machine Integrate with slack (write experiment results into new channel) Integrate with git TFS repository Test the training (with new commits into repo) Trained models should be saved locally in accenture machine (because of 300mb size)

ID	Title	Assigned To	State	Description
	It will be comfortable to have one.			
9771	As a developers, we want to have web socket server, so our flask server can call the websocket server, which will generate the models.	Andrej Pisarcik	Closed	When the school virtual machine starts, it needs to wait for connection from external agent (Accenture machine). When the request from Client pc comes, school virtual machine must ask the file from the Accenture Inference machine. When [WS Client] starts, try to connect to WS Service. (Repeat until success). When it connects, wait for request from school virtual machine. Respond to request with the static file. Port 443 on specified URI on the school machine. Create new repository for accenture machine.
10119	As a team, we must have sprint 1-3 documented, because we are obliged to for subject.	Marek Drahos	Closed	Inzinierske dielo https://docs.google.com/document/d/1eLUq6wszjWFBjZw_5UAGH2njD4jZa8jNh3fhg8XUvvl/edit Riadenie projektu https://docs.google.com/document/d/1khojBT5EtdC7H6Ue_zrEjvTCNEhnAEmZuigYh-I7a8XY/edit for more information see: http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/materialy/dokumentacia2017-18.pdf deviaty týždeň semestra (podľa rozvrhu stretnutí, pracovný deň nasledujúci po dni v týždni, kedy má tím plánované stretnutia), najneskôr však 23.11.2018, 12:00 (tí čo máte stretnutia v piatok, naplánujte si to tak, aby ste to zvládli) odovzdanie dokumentácie po prvých troch šprintoch (vrátane celkového obrázku projektu) plus dokumentácia k riadeniu - elektronicky * Úvod - o čom je tento dokument, ciele, ohraničenia. * Globálne ciele projektu na zimný semester * Celkový pohľad na systém (architektúra, dátový model, diagramy tried, moduly plus referencie/zoznam priložených e-dokumentov) Retrospektiva a aj vyjadrenie ako sprint prebiehal (čo sa spravilo, čo sa nedstihlo, pridať nejaký burndown chart, spraviť vyjadrenie či sme spravili toho viac alebo menej, info k organizácii timoveho projektu atd) http://www2.fiit.stuba.sk/~bielik/courses/tp-slov/tp-main.html#odovzd-zs
10122	As a user, I want to have inference be done on Accenture machine, because I want to generate new unique models.	Matej Mikus	Closed	When request is made, Neural network will choose correct model and generate object And returns that object 1 class (chairs) only for now
10125	As a user, I want to smooth out the generated 3D models, because it takes too long to render.	Marek Drahos	Closed	Research server side and research client side optimization Reduce the number of vertices Make parser better Find algorithm to smooth out and try to use it
10126	As a developers, we want to have methods for c# and python, because we want to be better organized.	Adam Popemik	Closed	As a developers, we want to have methods for c# and python, because we want to be better organized.
10127	As a user, I want to delete the object in the scene, because I did not like it.	Jan Antal	Closed	After pointing or grabbing the object, the user can delete the object with a button.

Export úloh piatego šprintu

ID	Title	Assigned To	State	Description
10123	As a developers, we want to have CI for websocket service. (accenture machine)	Jan Antal	Active	Setup the agent on accenture machine. (windows) Setup the CI from tfs git repository.
10124	As a developers, we want to have CI for our school virtual machine.	Marek Drahos	Active	When we commit new code into git, the program updates the VM of flask server. Another build and release.
10404	As a developers, we want to improve methods for c# and python, because we want to be better organized.	Adam Popernik	Active	As a developers, we want to have methods for c# and python, because we want to be better organized.
10407	As a developers, we want to have web socket server, so our flask server can call the websocket server, which will generate the models.	Andrej Pisarcik	Active	When the school virtual machine starts, it needs to wait for connection from external agent (Accenture machine). When the request from Client pc comes, school virtual machine must ask the file from the Accenture Inference machine. When [WS Client] starts, try to connect to WS Service. (Repeat until success). When it connects, wait for request from school virtual machine. Respond to request with the static file. Port 443 on specified URI on the school machine. Create new repository for accenture machine.
10410	As a user, I want to have inference be done on Accenture machine, because I want to geenerate new unique models.	Matej Mikus	Active	When request is made, Neural network will choose correct model and generate object And returns that object I class (chairs) only for now
10412	As a developer, I want to have a build/train/deploy pipeline for neural networks, because It will be comfortable to have one.	Peter Papay	New	https://github.com/IDSIA/sacred Setup the accenture machine Integrate with slack (write experiment results into new channel) Integrate with git TFS repository Test the training (with new commits into repo) Trained models should be saved locally in accenture machine (because of 300mb size)
10413	As a developers, we should add authentication to various parts of our pipeline, because we want to secure our network.	Dominik Mazak	Resolved	Setup basic authentication on apache (https) Implement in client the header with username/password - Initialisation of request
10513	As a developers, we want to fix the state of our project.	Peter Papay	Active	