

Arbeitsjournal STARHack

8. - 10. März 2019

David Schafer

2019-04-08

Inhaltsverzeichnis

1 Vorbereitung STARHack	2
1.1 Packliste	2
1.2 Git-Repository	2
2 Freitag 7. März 2019	3
2.1 - 14.00 Uhr - Anreise	3
2.2 - 16.00 Uhr - Anmeldung und Einrichten	3
2.3 - 19.00 Uhr - Case Präsentationen / Opening Ceremony	3
2.4 - 20.00 Uhr Brainstorming	4
2.5 - 22.00 Uhr Planung	4
2.6 - 23.30 Uhr Start Coding (till drop)	5
3 Samstag, 8. März 2019	6
3.1 - 06.00 Uhr - Schlafen	6
3.2 - 09.00 Uhr - Hacking / Coding	6
3.3 - 12.00 Uhr - Lunch	6
3.4 - 13.00 Uhr - Hacking / Coding	7
4 Sonntag, 9. März 2019	7
4.1 - 00.00 Uhr - Mitternachtssnack (Midnight Snack)	7
4.2 - 00.30 Uhr - Coding	8
4.3 - 08.30 Uhr - Coding	8
4.4 - 10.00 Uhr - Project Submission / Project Deadline	8
4.5 - 10.00 Uhr - Präsentationsvorbereitung & letzte Feinschliffe	8
4.6 - 10.50 Uhr - Präsentation Judging Pitches	9
4.7 - 14.00 Uhr - Closing ceremony	9
4.8 Abschliessende Worte	10

1 Vorbereitung STARTHack

1.1 Packliste

Freitag Morgen, 07. März - in wenigen Stunden beginnt mein erster Hackathon in St Gallen. Da stellt sich natürlich die Frage, was nehme ich mit?

Neben den normalen Utensilien wie Kleider, Necessaire und Schlafsack, wollte ich natürlich für jede Challenge gerüstet sein und packte noch folgendes ein:

- Surface Book
- 4 x 27" Monitor (dazu später mehr)
- Maus & Tastatur
- Raspberry Pi
- Steckbrett, Kabel & sämtliche IoT Sensoren (so ca. 20 Stück)
- Udoo X64
- 'Over the Ear' Kopfhörer
- Ersatz Kopfhörer (man weiss ja nie)
- Den Studentenausweis

Damit sollte ich eigentlich für alles gerüstet sein - egal welche Challenge kommen wird. Guten Mutes, alles dabei zu haben, verlies ich also gegen 14 Uhr die Wohnung Richtung St. Gallen und vergas prompt mein Ladegerät zu Hause.

1.2 Git-Repository

Vor dem Hack galt es noch zu bestimmen, wie wir miteinander kollaborieren werden. Schnell war klar, dass wir unseren Code mittels Git und Github verwalten werden.

Wir haben uns darauf geeignet mittels Gitflow zu arbeiten, damit jeder an seinem Part arbeiten kann und wir uns nicht ständig mit Konflikten auseinandersetzen müssen. In der Theorie klingt das gut, praktisch hat es leider nicht so geklappt. Wir haben uns gegenseitig einige Konflikte generiert, was wohl am akuten Schlafmangel lag.

Unser Github Repository: <https://github.com/tschibu/starthack-asimov>

2 Freitag 7. März 2019

Der Tag steht ganz im Zeichen von AI, Deep-Learning, Blockchain und noch mehr #Buzzwords.

2.1 - 14.00 Uhr - Anreise

Um 14:00 Uhr ging es Los. Mit vollbepacktem Auto Richtung STARTHack. Auf dem Weg haben wir schon viele Gedanken und Ideen in einem Brainstorming ausgetauscht. Neben den STARTHack Challenges gibt es immer die Möglichkeit ein eigenes Projekt zu verfolgen. Wir einigten uns darauf die Challenges abzuwarten bevor wir uns festlegen, den mit den aufgelisteten Partnern wie Microsoft, Swisscom & Bosch ist definitiv ein grosses Potential für spannende Projekte vorhanden.

2.2 - 16.00 Uhr - Anmeldung und Einrichten

Wir sind an der Uni St. Gallen angekommen und suchen zuerst mal einen Parkplatz, an dem wir unser Ganzes Material abladen können. Nachdem wir einen geeigneten "HSLU Ecken" an der Uni St. Gallen gefunden haben, ging es weiter zur Akkreditierung, wo wir unser Team registrieren konnten und unsere STARTHack Pässe erhielten. Als nächster Punkt auf der Tagesordnung stand die Opening Ceremony sowie die Präsentation der Challenges. Bis dahin, richteten wir unsere Arbeitsplätze noch optimal ein. Bildschirm, Maus und Tastatur werden installiert. Der Raspberry bleibt vorerst noch in der Tasche.

2.3 - 19.00 Uhr - Case Präsentationen / Opening Ceremony

Im grossen Hörsaal der HSG ging es dann los mit der Opening Ceremony. Angefangen mit einer sehr Interessanten Key Note von Damion Borth (aka Germanys Mr. Deep Learning) über Ai, Machine Learning und neuronale Netze. Danach ging es los mit der Präsentation der 8 verschiedenen Challenges:

- Microsoft: Microsoft AI
- Volvo: Car Sharing App
- Autosense: Generating and Visualizing Crash Data
- Leica: Accurate Augmented Reality
- FLETA: Blockchain

- INVENTEX: Trust Systems
- SBB: Coffee Cup Recycling
- BOSCH: New Car Features

Die detaillierten Case Beschreibungen findet man hier: <http://live.STARThack.ch/case-descriptions/>

2.4 - 20.00 Uhr Brainstorming

Wir hatten nach den kurz Präsentationen noch Zeit 2 Cases im Details anzuhören. Wir entschieden uns für autoSense und BOSCH.

Im **BOSCH** Case ging es darum, neue Features für ein "modernes" Auto zu entwickeln, welche die Umweltbilanz, Sicherheit und/oder Gesundheit des Fahrers verbessern sollen. Dabei stellte BOSCH ein Auto zur Verfügung, bei welchem Sensor Daten über eine API abgefragt werden konnte. Ausserdem bestand die Möglichkeit ein Tablet sowie die Stereoanlage im Auto zu manipulieren - mehr jedoch nicht. Ein kontrolliertes abbremesen des Fahrzeuges bei einem Sekundenschlaf wäre nicht möglich über die angebotene API.

Im **Autosense** Case ging um eine SensorBox welche via OBD Sensor-Daten vom Auto ausliest. Zusätzlich besitzt die Autosense-OBd-Box noch eigene Sensoren wie ein Accelerometer, welches als "Crash Sensor" fungiert. Die Challenge bestand darin, mithilfe der Sensor Daten zu berechnen wo im Auto der Einschlag, bei einem Unfall, war. Das ganze sollte danach als 3D Modell visualisiert und via REST-Schnittstelle zur Verfügung gestellt werden.

Bis 22 Uhr hatten wir Zeit uns für ein Thema zu entscheiden. Aufgrund des Mathematischen Charakters wählten wir die **Autosense** Challenge und erhielten die Bestätigung um 22:00 Uhr - es geht los.

2.5 - 22.00 Uhr Planung

Zuerst ging es mal um die Planung. Wir überlegten was alles zu tun ist und was die besten Mittel zum Ziel sind. Wir entschieden uns, dass sich jeder zuerst eine halbe Stunde in die Materie einliest und wir uns danach für ein Stand-Up treffen.

Wir waren uns zum Glück schnell einig, was für Komponenten für die Lösung notwendig sind. Im Stand-Up definierten wir dann auch gleich die einzusetzenden Werkzeuge/Libraries:

- Daten Transformation: JSON Parser, Mathematische Berechnungen

- Webserver für REST API: Sanic
- Visualisierung der Daten: OpenCV
- Cloud Kompatibilität: Docker

Danach teilten wir die Aufgaben auf. Steve kümmerte sich um den Webserver, Remo um die Visualisierung und Serge und ich um den JSON Parser und die Mathematischen Berechnungen.

2.6 - 23.30 Uhr Start Coding (till drop)

jetzt ging es also Effektiv los mit dem Coding. Jeder erstellte im Git Development Branch ein Feature und begann zu Coden.

Ich schaut mir zuerst mal das JSON genau an, um herauszufinden was für Daten relevant sind für die Berechnung und wie sie codiert sind. Dabei sah ich schon die erste Hürde, die ganzen Daten sind unleserlich: Base64 Codiert.

Zuerst erstellte ich also eine Hilfsfunktion zum dekodieren auf UTF-8. Danach konnte ich im Parser die Daten weiter analysieren. Vor allem die Beschleunigungsdaten (x, y und z-Achse), sowie die Referenzwerte zu g (Anziehungskraft) und der sogenannte "Time Offset" - also die relative Zeit, zum Start des Crash-Recording, an dem eine spezifische Beschleunigung gemessen wurde - waren wichtig.

Ca. um 02:30 Uhr haben Serge und ich uns abgesprochen, schon etwas müde entschieden wir uns, dass wir die Offset Zeiten unbedingt in reale Zeiten umwandeln müssen für eine akkurate Berechnung. Das beschäftigte uns bis ca. 03:30 Uhr und stellte sich dann als absolut nutzlos heraus - zurück auf Anfang also.

Die Offset Zeiten belassen wir nun wie sie waren, nach einem kurzen Stand-Up definierten wir, was effektiv wichtig ist für die Berechnung des Einschlagwinkels und der Einschlagkraft:

- Die Beschleunigungen x und y-Achse
- Umwandlung der Beschleunigungen in g (mit Hilfe des g-Referenz Parameters)
- Winkel des Einschlags mittels Vektorgeometrie

Um 04:30 waren wir soweit bereit mit dem Parser. Alles relevante wurde umgerechnet, nun ging es um die Mathematik. Nach Bald 24h ohne Schlaf war mein Gehirn aber nicht mehr in der Lage für simple Mathematik, sämtliche Berechnungen die ich bis 06:00 gemacht habe waren falsch.

Um 06:00 war es dann soweit und mein Körper verlangte dringend Schlaf. Ich legte mich auf die Luftmatratze neben unserem Pult und schlief für ein paar Stunden.

3 Samstag, 8. März 2019

3.1 - 06.00 Uhr - Schlafen

Schlafen war etwas gewöhnungsbedürftig, das Licht brannte und um uns herum war voller Betrieb. Viele Leute waren noch wach oder bereits wieder wach. Also hiess es Kopfhörer rein, Decke über den Kopf und versuchen weiter zu schlafen.

3.2 - 09.00 Uhr - Hacking / Coding

Wie wunderbar, was 3 Stunden schlaf bewirken können! Als ich aufwachte war ich zwar nicht wirklich erholt aber mein Kopf funktionierte wieder. Nach dem Frühstück sah ich mir meine Berechnungen vom Vorabend an, schmunzelte über mich selbst und machte die Berechnungen nochmals neu. Diesmal stimmten sie.

Der Trick war, die Euklidische Distanz mittels x und y Kräften zu berechnen, so konnte schlussendlich auf den Einschlagwinkel abgeleitet werden.

Der erste Testrun ergab aber merkwürdige Resultate, da sah ich dass die Beschleunigungswerte von autoSense nicht ohne weiteres umgerechnet werden konnten. Es musste zuerst mittels einer weiteren Hilfsvariable (Calibration genannt) umgerechnet werden. Denn die Beschleunigungswerte sind auch nur "relativ" zur am Anfang gemessenen Calibration.

3.3 - 12.00 Uhr - Lunch

Um 12:00 Uhr ging es dann zum Lunch, Spätzlipfanne mit Speckwürfeln. Übrigens, was die Ganze Verpflegung und die Organisation rund um den Hackathon anbelangt: Respekt.

Rund um die Uhr war der hilfsbereite Staff vom STARTHack anwesend. Überall standen Kühlschränke mit Getränken. Und es gab sogar separate, immer volle, RedBull Kühlschränke (ich glaube ich habe 20 RedBull an diesem Wochenende getrunken).

3.4 - 13.00 Uhr - Hacking / Coding

Nach der Mittagspause ging es dann weiter. Ich vervollständigte meinen Teil des Parsers sowie die Berechnungen. Der Webserver schien soweit auch schon fast bereit zu sein.

Gegen 17 Uhr funktionierten dann die Berechnungen soweit zuverlässig. Ich konnte dann noch Remo unterstützen, um via OpenCV eine Grafik der Einschlagkraft und des Winkels zu berechnen. Auch hier war wieder etwas Mathematik notwendig, um aus dem Winkel einen Pfeil zu machen. Auch hier wieder, Trigonometrie dein Freund und Helfer. Mittels Sinus und Cosinus sowie einem "virtuellen" Punkt in der Bildmitte konnten wir akkurat den Einschlagwinkel visualisieren. Bei der Visualisierung der Einschlagkraft entschieden wir uns für einen Ring, welcher je nach Grösse der Einschlagkraft (in unseren Testdaten zwischen 2 und 11g) grösser wurde.

Nach dem Abendessen ging es dann um die Kombination aller Teilsysteme. Um ca. 22 Uhr hatten wir dann ein funktionierendes Docker Image, welches die Daten als JSON entgegennimmt, berechnet und visualisiert.

Soweit waren wir also eigentlich fertig, ein gutes Gefühl. Da wir uns aber nicht mit der Grundfunktionalität alleine Zufrieden geben, definierten wir noch die "Stretch Goals":

- HTML/CSS Website
- JSON Daten via Drag&Drop einlesen
- Weitere nützliche Infos (Crash Date, Offset...)
- Azure Webservice

Um ca. 2:00 Uhr hatten wir eine halbwegs vernünftige Website kreiert. Die Müdigkeit war bei uns allen aber so gross, dass wir es für heute gut sein liessen und uns Schlafen legten.

4 Sonntag, 9. März 2019

4.1 - 00.00 Uhr - Mitternachtssnack (Midnight Snack)

Ein Mitternachtssnack! Bratwurst und Brot - nicht wirklich optimal für unsere Vegetarischen Kollegen. Und ich selber hatte keinen Hunger (evtl. wegen den 20 RedBull?). Eine kurze Pause an der frischen Luft war aber sehr willkommen, danach ging es dann weiter.

4.2 - 00.30 Uhr - Coding

Wir arbeiten bis ca. 02:00 an den Stretch Goals, ich versuchte vor allem noch den Code zu optimieren (etwas Clean Code Praktiken umsetzen) und den Azure Webservice vorzubereiten.

Um ca. 2:00 Uhr hatten wir eine halbwegs vernünftige Website kreiert. Die Müdigkeit war bei uns allen aber so gross das wir es für heute gut sein liessen und uns Schlafen legten.

4.3 - 08.30 Uhr - Coding

Nach ca. 7h erholsamen Schlaf, der bitter nötig war, gingen wir Frühstück und besprachen was noch zu tun war.

Bis ca. 10 Uhr hatten wir noch Zeit für den letzten Feinschliff unseres Codes.

4.4 - 10.00 Uhr - Project Submission / Project Deadline

Um 10 Uhr war dann die Deadline angesetzt. Ich zippte unseren Code und schrieb noch rasch eine ansprechende Beschreibung unserer Lösung und submittete um 09:55.

Der DevPost kann hier gelesen werden: https://devpost.com/software/autosense_group_asim

4.5 - 10.00 Uhr - Präsentationsvorbereitung & letzte Feinschliffe

Nun ging es noch um die Präsentation, beziehungsweise das Pitching unserer Lösung. 1h Vorbereitungszeit scheint zwar recht wenig zu sein, aber da wir unsere Lösung ja in und auswendig kannten reichte die Zeit (knapp) aus. Wir stellten eine kleine Powerpoint Präsentation zusammen und stellten den Betrieb unserer Demo Website sicher.

4.6 - 10.50 Uhr - Präsentation Judging Pitches

Um 10:50 war es dann soweit. Wir traten gegen 5 andere Teams an. Vor allem Team Nr. 1 verunsicherte mich bereits stark. Ihre Lösung beinhaltete das glätten der stark rauschenden Beschleunigungsdaten. Etwas an das wir nicht mal im entferntesten gedacht haben. Bei der Demo Ihrer Lösung fiel mir aber auf, dass die Einschlagwinkel sehr merkwürdig scheinen.

Unbeeindruckt davon präsentierten wir aber unsere Lösung und die Judges schienen wirklich beeindruckt zu sein. Alles in allem verlief der Pitch gut. Einzig bei der Demonstration funktionierte ein "Last Minute" Feature nicht wie es sollte (Ein Schiebebalken um den Zeitverlauf der Kräfte anzuzeigen). Das merkte man auch an der Reaktion der Judges, hätten wir dieses Feature nicht gezeigt wäre unsere Lösung wohl noch besser angekommen.

Neben unserer Gruppe präsentierte ebenfalls noch ein befreundetes HSLU Team (das übrigens den ganzen Hackathon neben uns sass) ihre Lösung. Ihre Lösung war natürlich sehr ähnlich wie unsere da man sich unter Freunden auch mal austauscht hat ;) Jedoch funktionierte Ihr Feature mit dem Schiebebalken einwandfrei - kostet uns das den Sieg?

Am Schluss der Pitches gaben die Judges noch bekannt wie die Sieger erkoren werden. Wer die meisten Visualisierungen so hatte, wie die Crashes in Wirklichkeit abliefen hat gewonnen. Am Schluss zeigten sie uns noch die Bilder der echten Crashes. Eine kurze Kopfrechnung ergab das wir (und das andere HSLU Team) ca. 2 - 3 der 5 Crashes richtig darstellten - Mit Linear Beschleunigung zu rechnen ist wirklich keine Freude.

Nach den Judging Pitches räumten wir unsere Plätze auf und warteten auf die Siegerverkündung, welche auf 14:00 angesetzt war. Wir entschieden uns aber dafür die Ending Ceremony via Livestream zu verfolgen und uns bereits auf den Weg Richtung Luzern zu machen.

4.7 - 14.00 Uhr - Closing ceremony

Um 14:00 Uhr war es dann soweit. Zuerst zeigte natürlich das STARTHack Team einige Impressionen bedankte sich bei allen Partnern. Dann ging es zur Siegerehrung über - leider erhielten wir keinen detaillierten Einblick in die Arbeiten der anderen Teams, was etwas schade war. Einzig 3 andere Teams stellten Ihre Lösung kurz vor, interessant fand ich das Sieger Team von Bosch, die eine sehr kreative Lösung präsentierten:

<https://www.youtube.com/watch?v=dRYWBtBHoKo>

Nun war Autosense an der Reihe. Voller Spannung erwarteten wir die Siegerverkündung. Zwei Teams waren sehr nahe beieinander heisst es. Und das Sieger Team ist... nicht unser Team asimov. Aber das Team ipv5, unsere HSLU Kollegen, welchen wir den Sieg natürlich genau so gönnen!

Und ein solider zweiter Platz ist auch vollkommen in Ordnung.

Am Schluss wurden nochmals einige Infos abgegeben zum nächsten START-Hack 2020.

4.8 Abschliessende Worte

Der Hackathon war abschliessend gesagt, eine sehr interessante Erfahrung für mich. Vor allem das Arbeiten im Team, mit einer engen Deadline und einer einzigartigen Challenge war sehr spannend.

Unser Team hat gut funktioniert, die Aufteilung der Arbeiten war effizient und es kam zu (verhältnismässig) wenigen Konflikten beim Coding.

Als Lektion für das nächste mal würde ich aber definitiv mehr Zeit ins Brainstorming und die Planung investieren. Hätten wir zu Beginn mehr Zeit investiert, wären wir viel schneller ans Ziel gekommen und hätten mehr Zeit für unsere Stretch Goals investieren können.

Das war nun mein erster Hackathon und die Erfahrung war einmalig. Und das war definitiv nicht mein letzter.