feedzai

2022 Feedzai Summer Internships

Feedzai is a software house company that develops a fraud detection product used by some of the largest banks and online retailers in the world. In the summer of 2022 we will have several paid summer internships available for students in Portugal.

feedzai

Cycle detection for CM Automations Rules

Original proposal by: Luís Costa | Type: Backend/Java | Preferential location: Lisbon

Case Manager allows its users to define automatic actions that are executed given some criteria. These are known as Automations Rules. They are event-driven rules that will be triggered by events happening in CM and, as a result, new actions will be executed if those events fulfill the rules' conditions. Those rules can be chained with each other in order to compose more complex event-driven orchestrations.

However, when the net of interconnected rules becomes too complex, it's possible, or even likely, that the triggered automations enter a loop (ie. rules that feed each other indefinitely).
This is a production hazzard that leads to impactful production issues.
The goal for the internship is to investigate a method to detect loops, either in execution or at configuration time, making the system resilient.

Keywords: event-driven rules, direct graph modeling, cycle detection, software resilience
Number of internships: BE x 1

Internship goals:
1. Use directed graphs to model CM Automations Rules,
2. Identify cycles on those directed graphs,
3. Identify the Automations Rules are creating the cycle (to prevent cycles from being deployed),
4. Define a strategy to break cycles from rules already deployed (recover from cycles that are already deployed),
5. Implement a functional PoC,
6. Stretch: PoC with no breaking changes

Requirements:

- Knowledgeable of Directed Graphs' concepts and algorithms
- Some proficiency with Java8+
- Familiar with OOP concepts
- Eager to learn

Internship plan (12 weeks):

2 w. Onboarding: env setup, company onboarding, CM/team onboarding
1 w. Design the PoC and define the statement of work
2 w. Define and implement a representation of ARs' using directed graphs
2 w. Define and implement an algorithm to detect cycles on ARs' DG
1 w. Refine the above algorithm to identify the beginning/end (rules) of the cycle
2 w. Define and implement a strategy to break from cycles
1 w. Testing and validation
1 w. Wrap up and final presentation

Notification Center

Original proposal by: José Sousa | Type: Frontend/React, Backend/Java | Preferential location: Lisbon

RiskOps Studio is one of the newest products currently being developed in Feedzai. It tries to integrate the existing products into a single application to create a consistent, truly unified experience where the user can access all features without the need to open an extra tab or window in its browser. One of the features that all existing products lacked until now was a way to notify our users. Maybe an asynchronous task finished or a user wants to notify another that an action is required by him in another context.

All of this is not possible if we don't have a notification center. The main challenges of this project are:

- Discover how to handle the communication between the different micro services (backend & frontend);
- Define and build Notification Center API (backend & frontend);
- Add support in the existing products for the Notification Center (backend & frontend);
- Build the Notification Center UI (frontend);

Number of internships: 1xBE & 1xFE

Frontend

Keywords: React, Micro Frontends, Webpack, Cypress, Storybook, Typescript.

Requirements:

- Knowledge about the Web Platform features;
- Knowledge about Javascript & CSS;
- Some experience with React;
- Eager to learn;
- Typescript, Storybook and Cypress are a plus;

Internship Plan (~12 weeks):

- Onboarding; (2 weeks)
    - Computer setup;
    - Tutorials (Javascript, React, Typescript, Micro Frontends or any other technologies/concepts needed);
- Designing a PoC; (2 weeks)
- Building the Notification Center API (shared between the multiple Micro Frontends) (2 weeks)
    - Aligning the Notification Center API with the Backend Services;
- Building the Notification Center UI in RiskOps Studio; (2 weeks)
- Integrate the Notification API in Feedzai products; (3 weeks)
    - Create a client to ease the integration process;
- Presentation & Documentation (1 week)

Backend

Keywords: Java, Kubernetes, Quarkus, Sockets, Micro services.

Requirements:

- Some experience with OOP and Java is a plus.
- Eager to learn and grow.

Internship Plan (12 weeks):

- Onboarding; (2 weeks)
    - Computer setup;
    - Tutorials (Java 17, Quarkus, Docker, Kubernetes or any other technologies/concepts needed);
- Designing a PoC; (2 weeks)
- Building the Notification Center API (4 weeks)
    - Aligning the Notification Center API with the Frontend Services;
- Integrate the Notification API in Feedzai products; (3 weeks)

    - Create a client to ease the integration process;
- Presentation & Documentation (1 week)

Template pages

Original proposal by: João César | Type: Frontend/React | Preferential location: Lisbon

RiskOps Studio is one of the newest products currently being developed in Feedzai. It tries to integrate the existing products into a single application to create a consistent, truly unified experience where the user can access all features without the need to open an extra tab or window in its browser.
Most of our CRUD pages have the same structure and should:

- Have a consistent UI between the multiple modules (RiskOps Studio, Case Manager, Pulse, etc);
- Reuse all shared logic like: data fetching, filtering, triggering actions, etc;
- Handle errors the same way;

Number of internships: 1xFE
Keywords: React, Micro Frontends, Webpack, Cypress, Storybook, Typescript.

Requirements:

- Knowledge about the Web Platform features;
- Knowledge about Javascript & CSS;
- Some experience with React;
- Eager to learn;
- Typescript, Storybook and Cypress are a plus;

Internship Plan (~10 weeks):

- Onboarding; (2 weeks)
    - Computer setup;
    - Tutorials (Javascript, React, Typescript, Micro Frontends or any other technologies/concepts needed);
- Accessing the requirements and current capabilities of existing products; (2 weeks)
- Designing a PoC; (3 weeks)
- Integrate the PoC in Rules project (2 weeks)
- Presentation & Documentation (1 week)

feedzai

Audit

Original proposal by: José Sousa | Type: Backend/Java, Frontend/React | Preferential location: Lisbon

Feedzai works with money, which means everything that happens in our systems must be traceable. Every important action performed by the users of our systems must be tracked so that everything can be explained and, if necessary, reverted. Currently, our products do this independently, each with its own solution. Our vision is to centralize this audit process into a single, multi-tenanted, cloud-native component. In this internship, you will be responsible for developing this new solution using the latest cloud-native technologies, such as Kubernetes. At the end, you will have developed a modern API that is capable of auditing actions from any of our current and future products, in a way that is scalable and cost-efficient.

Number of internships: 1xBE & 1xFE

Frontend

Keywords: React, Micro Frontends, Webpack, Cypress, Storybook, Typescript.
Requirements:

- Knowledge about the Web Platform features;
- Knowledge about Javascript & CSS;
- Some experience with React;
- Eager to learn;
- Typescript, Storybook and Cypress are a plus;

Internship Plan (~12 weeks):

- Onboarding; (2 week)
    - Computer setup;
    - Tutorials (Javascript, React, Typescript, Micro Frontends or any other technologies/concepts needed);
- Setting up a new Micro Frontend and integrate it in RiskOps Studio; (1 week)
- Designing the Audit page; 6 weeks)
- Integrate the Audit page with the Backend API services (2 week)

feedzai

Backend

Keywords: Java, Kubernetes, Quarkus, Micro services.
Requirements:

- Some experience with OOP and Java is a plus.
- Eager to learn and grow.

Internship Plan (12 weeks):

- Onboarding; (2 week)
  - Computer setup;
  - Tutorials (Java 17, Quarkus, Docker, Kubernetes or any other technologies/concepts needed);
- Get familiar with the current implementation and revamp it/design the API; (3 weeks)
- Implement the new audit API (4 weeks)
- Integrate with FE (2 weeks)
- Presentation & Documentation (1 week)

Batch Processor Revamp

Original proposal by: Pedro Faria | Type: Backend/Java | Preferential location: TBD

Batch Processor is a Feedzai "micro service" developed in-house which reads file-fased batches of data and distributes it to our scoring engine. Unlike the rest of the architecture, Batch Processor is not horizontally scalable which limits its exit throughput. This initiative would look into solving this architectural bottleneck, with an eye on performance and reliability. The student would also be looking into adopting industry standard technologies that could be used to fundamentally refactor Batch Processor, if that move is demonstrably a) cost-effective, b) fit for purpose.

The objectives of this internship will be to propose a solution that will:

- Integrate into current Feedzai platform;
- Enable large scale data processing;
- Observe the principles of Scalability and Reliability;
- Be aligned with industry standards and latest trends;
- Cloud Native;

Number of internships: 1xBE
Keywords: Batch Processing, High Volumes of Data, Data Processing, Cloud Computing, Scalability
Requirements:

- Knowledge/curiosity about Batch Processing and Streaming Processing;
- Knowledge/curiosity about ETL (ELT) patterns;
- Knowledge/curiosity about Cloud Computing;
- Experience in one of Java, Python, Scala or R
- Eager to learn;
- Eager to solve a complex problem;
- Spark, Spring Batch, AWS Batch or other batching or streaming frameworks are a plus;

Internship Plan (~10 weeks):

- Onboarding; (1 week)
    - Computer setup;
    - Problem Statement;
    - Current Solution (Batch Processor) Pains and Constraints;
- Investigation of current market best practices and de-facto frameworks (1 week)
- Brainstorm and Idealization (2 weeks)
- Designing a PoC; (3 weeks)
- Performance Testing and Benchmarking (2 weeks)
- Presentation & Documentation (1 week)

Product Demos Dashboard

Original proposal by: José Coelho | Type: Cloud & Frontend | Preferential location: Porto

There are, on average, about 60 product demo applications running on the AWS demos account. These demos are used by many different teams, from sales and pre-sales teams to solutions and product teams, for various demoing purposes.
With such a big number of different environments and stakeholders, we need a central place where anyone can check the status and information about their environments.
The purpose of this internship is to build a cloud-native web application for a dashboard where people can find a list of all the demos available and then drill down into a more detailed view of their specific environment. There is a legacy version of this dashboard, the

idea would be to revamp it and adapt it to the new demos deployment model. You can check it out here

Requirements

1. General software development knowledge and willingness to learn
2. Frontend development experience is a plus
3. Humble and easy to work with

Internship plan (10 weeks)

1. Onboarding - 1 week
2. Designing the application - 2 weeks
3. Implementing the application - 6 weeks
4. Showcase, document and educate on the Product Demos Dashboard - 1 week

Automagic analysis of test coverage by feature

Original proposal by: Bernardo Maciel and Ricardo Lopes | Type: QA

Code coverage is a metric computed easily in software engineering and it allows us to understand which part of our system is covered by unit tests. Test coverage is a black-box testing technique that maps the coverage of requirements outlined in multiple documents against the tests that have been executed. Unlike code coverage, test coverage is not so simple to determine.

At Feedzai, we want to guide our test coverage according to the features we develop, and so we need to:

- understand what a feature is
- which features a product has
- which tests cover its functionalities

in order to establish a traceability link between features and our automated tests.

Doing this manually is time consuming and error prone. But having an automated way to track how our features are covered in terms of automated tests is very valuable. This analysis provides insights over the status of the (UI, system and integration) test automation suites of a product, such as coverage gaps.

The objective of this internship is to design and implement a tool that could automatically build and update a bi-directional traceability matrices between features and (UI, system and integration) automated tests that would allow the team to continuously track the set of test cases that cover our features and enforce test categorization practices in our codebases.

Requirements:

- Programing skills
- Ability to tackle complex problems
- Experience in Java and JUnit
- Experience in Javascript is a plus

The internship plan (total of 8 weeks) is:

- Company onboarding (1 week)
- Analyze the problem of automatic test coverage, including existing approaches to the problem at Feedzai (1 week)
- Design the tool (3 weeks)
- Apply the tool in one of the Feedzai repositories (1 week)
- Write an adoption guide to drive adoption in other Feedzai repositories (1 week)
- Present the results by doing an internal presentation (1 week)

Sources:

- Using Feature-Driven Testing for More Strategic Test Coverage
- Code Coverage vs Test Coverage : A Detailed Guide
- JUnit Categories

Risk Studio Dashboard/Landing Page(s)

Original proposal by: Sofia Carvalho | Type: Product/UX Designer | Preferential location: TBD

Feedzai's Risk Studio allows our clients to review alerts regarding transactions, in order to catch fraud as quickly as possible and protect end-users. It's also used by investigators to discover and prevent new fraud patterns, and dynamically create rules and automations to stop fraud in its tracks.
Feedzai can have different types of users and roles using Risk Studio (from L1 analysts to managers regarding different use cases like Fraud and AML - Anti Money Laundering -, ranging from tier-1 Banks to Merchants) and is part of our goal to make them as effective

and efficient as they can, while allowing them to access the most relevant data to perform their daily tasks and workflow.

The goal of this internship is to deliver a design adapted for these different roles and needs, while going through the entire UX process: from discovery to high-fidelity mockups, socializing with internal stakeholders (Engineering, Product Managers, Subject Matter Experts, Data Visualization Engineers and UX Design Team) and getting insights from external users, collecting metrics and data that will allow you to define the final WGLL design (*what great looks like*). The final result will be a landing page with the proposed solution that should be available to Investigation and/or Strategy users' dashboards. You'll also be able to set metrics that will later on be retrieved by our analytics product to understand the impact of your designs in our users' day-to-day workflow.

In this internship you will make a significant contribution to an application used worldwide to protect millions of people and their money (probably yours too!) and you will work closely with our highly-skilled product team to make a significant improvement to our application.
Cherry on top of the cake? You'll be able to add this project to your portfolio and get a glimpse on what it is like to work as a designer in a product (and unicorn) company.

Requirements

- Excellent communication skills with the ability to interact with different stakeholders
- Good understanding of lean UX practices and processes, including user-centered design and research
- Good understanding of relevant standards and best practices for design of modern, data-rich, and responsive applications
- Experience working on all aspects of the UX process - from defining user needs, to developing concepts, to validating designs, to reviewing implemented designs to ensure pixel-perfect implementation
- Nice to have: Strong UI / research skills and experience working with Figma

Internship Plan (12 weeks - 3 months)

1. **Onboarding:** Learn about Risk Studio and get familiar with UX tools and methodologies (1 week)
2. **Discover:** set a discovery plan, success metrics, perform interviews and delivery a research report (4 weeks)

3. **Define:** map the user journey, define 1+ user flows and present 2+ scenarios (3 weeks)
4. **Design:** present a proposal starting with paper prototyping and deliver high-fidelity mockups using the UI library available (2 weeks)
5. **Validate:** validate your proposal with final users and present a User-Testing Report (1 week)
6. **Document** and present the final solution to final stakeholders and write a short blogpost for Feedzai's Techblog sharing your experience and case study (1 week)

Revamp the definition of model explanations

Original proposal by: Marta Santos | Type: Frontend & UX | Preferential location: TBD

One of the core pillars of responsible AI is model explainability. Although it is already possible in our product for data scientists to set up explanations when building a model, it is cumbersome to manage in the long run. For this internship we are proposing the revamp of the model explanations configuration page to allow our customers to define explanations effortlessly, bringing their AI processes one step closer to the standards of responsible AI.
You'll be working closely with the product management, UX and FE team to define and implement a better user experience in our product.

Requirements:

- Experience in modern Javascript, HTML5 and CSS web standards
- Knowledge about React
- Testing frameworks like Jest or Cypress
- Willingness to learn more about the basis of a good user experience

The internship plan (total of 12 weeks) is:

- Understanding the problem by talking to product management and internal clients (1 week)
- Together with the UX and product management team create mockups for the new user experience (2 weeks)
- Gain context on how to implement in Pulse Frontend (2 weeks)
- Implement mockups (6 weeks)
- Present final result to relevant stakeholders (1 week)

feedzai

Smart data sampling

Original proposal by: David Santos | Type: Data Science | Preferential location: TBD

Feedzai provides a tool to sample data in a smart and distributed way for our customers, particularly those with very large datasets. This allows our clients to develop their machine learning models with a manageable amount of data and achieve similar performances as with the complete datasets. We aim to expand the support of this tool to cover more complex use-cases. This package is developed using Python and PySpark.

Requirements:

- Python proficiency
- Knowledge of Data Science concepts
- Knowledge of git and distributed computing are a plus

The internship plan (total of 12 weeks) is:

- Understanding the business context and how the Smart sampling fits (1 week)
- Install and gain knowledge on how to develop code on the DS-API (1 week)
- Learn Spark basics (1 week)
- Design, develop and test implementation (6 weeks)
- Test implementation scalability (2 weeks)
- Present final result to relevant stakeholders (1 week)

Scalable data exploration report

Original proposal by: David Santos | Type: Data Science | Preferential location: TBD

At Feedzai, one of the fundamental tasks of Data Scientists is to explore data with a variety of tools, goals and perspectives. We have an exploration package, developed in Python, that allows Data Scientists to easily perform complex data exploration. However, as datasets grow big (billions and above), this package suffers from scalability limitations. The proposal for this internship is to bring scalability to this package by identifying bottlenecks and improving job distribution.

Requirements:

- Python proficiency
- Knowledge of Data Science concepts

- Knowledge of git and distributed computing are a plus

The internship plan (total of 12 weeks) is:

- Gain context on what the data exploration is and supports (2 week)
    - What are the data exploration notebook functionalities?
    - Make it work on small data;
    - Make it fail on big data;
- Install and gain knowledge on how to develop code on the DS-API (1 week)
- Design, develop and test a solution (7 weeks)
- Test it on big data (1 weeks)
- Present final result to relevant stakeholders (1 week)

AutoML semantic configuration

Original proposal by: Marta Santos | Type: Frontend & UX | Preferential location: TBD

AutoML is a tool that automates the steps needed to create a baseline machine learning model. From feature creation and selection to choosing the best model hyper-parameters. As for any model, one of the most important steps is to understand the semantics of the fields, so that tailored features can be created and the best results are achieved. Currently, that semantic meaning is passed to AutoML through a configuration file in JSON format, which is a bit heavy on the eye!
You'll be working closely with the product management, UX and FE teams to define and implement a screen that would allow the user to define what was previously done in a file.

Requirements:

- Experience in modern Javascript, HTML5 and CSS web standards
- Knowledge about React
- Testing frameworks like Jest or Cypress
- Willingness to learn more about the basis of a good user experience

The internship plan (total of 12 weeks) is:

- Understanding the problem by talking to product management and internal clients (1 week)
- Together with the UX and product management team create mockups for the new user experience (2 weeks)
- Gain context on how to implement in Pulse Frontend (2 weeks)

- Implement mockups (6 weeks)
- Present final result to relevant stakeholders (1 week)

Benchmark fuzzy matching algorithms

Original proposal by: Rui Cachorreiro | Type: Data Science | Preferential location: TBD

In Financial Crime, more often than not, criminals will try to circumvent protections that are already in place. This includes writing addresses, names, and other attributes with typos intentionally, hoping the defense systems will not catch them. When they notice that "Street Fraud-a-lot" is being blocked when trying to open a bank account with stolen information, they will surely try "St Fraud-a-lottt" or even "Fraudalot St...,.,..|||#" not to mention other more creative names.

In this internship we aim to benchmark different **fuzzy matching algorithms** so that we make criminals sweat in despair by catching all their shady attacks.

Requirements:

- Knowledge of Data Science concepts
- Python proficiency
- Comfortable around the Python scientific stack (scipy, numpy, pandas, ...)
- Some experience of NLP (Natural Language Processing) is a plus.

The internship plan (total of 12 weeks) is:

- Understanding the problem by talking to product management and internal clients (1 week)
- Together with the AI Architects and Research teams investigate the state-of-the-art on fuzzy matching and deduplication algorithms (2 weeks)
- Design an experiment to benchmark said algorithms (2 weeks)
- Implement and execute benchmarking experiment (6 weeks)
- Present final result to relevant stakeholders (1 week)

feedzai

Forecasting alert volume

Original proposal by: Jorge Teixeira | Type: Data Science | Preferential location: TBD

One of the challenges of complex systems is how to monitor them and make sense in the middle of all the signals that are produced. At Feedzai we allow our customers to create complex risk strategies that culminate in great fraud detection or anti-money laundering results. However, our customers need to anticipate peaks or valleys of criminal activity so that they can adapt their operational capacity.

In this internship you will work closely with our Product and Research teams to develop a way to forecast alert volume as well as system performance variations, based on system history.

Requirements:

- Knowledge of Data Science concepts
- Python proficiency
- Comfortable around the Python scientific stack (scipy, numpy, pandas, ...)
- Some experience of forecasting in time-series is a plus.

The internship plan (total of 12 weeks) is:

- Understanding the problem by talking to product management and internal clients (1 week)
- Together with the AI Architects and Research teams create a technical requirements document (2 weeks)
- Gain context on how to implement the solution within Feedzai tools (2 weeks)
- Implement a MVP (6 weeks)
- Present final result to relevant stakeholders (1 week)

Pulse real-time notifications

Original proposal by: Tiago Martins & Pedro Moura | Type: Backend & Frontend | Preferential location: Porto

Pulse is one of the core systems at Feedzai, it allows users to design, maintain and run their risk strategy. Unfortunately the main approach to show users information in real-time has

been using a polling mechanism which is inefficient and doesn't offer the user a true real-time experience.
This initiative would be to migrate some of the Pulse features from polling to websockets in order to improve performance and give the user a better experience.

Number of interns: 1xBE & 1xFE
BE Requirements:

- Some experience with OOP and Java is a plus
- Knowledge or curiosity about real-time notifications
- Eager to learn and solve problems
- Willingness to work with other people to achieve the best outcome

FE Requirements:

- Experience in modern Javascript
- Knowledge about React
- Testing frameworks like Jest, Enzyme
- Knowledge or curiosity about real-time notifications

Internship Plan (10 weeks):

- Onboarding, Setup and deep dive on Pulse (2 weeks)
- Concurrent edit Feature analysis (1 week)
- Migrate polling requests and other events to web-sockets(2 weeks)
- Migrate Job Executions polling to web-sockets (3 weeks)
- Prepare Demo and present final results (1 week)
- Article about internship experience (1 week)

Optimizing Railgun

Original proposal by: Luís Alves | Type: Backend | Preferential location: Lisbon/Porto

Railgun is a new distributed system that we are developing at Feedzai. Its sole purpose is to compute hundreds of real-time metrics based on the transactions that we process, that are later used to make decisions regarding these transactions, e.g. "total amount spent by a given card over the last 1 year". It is designed to process thousands of transactions per second at low latencies (less than a second).

You can read more about Railgun in the following papers/blog posts:

- Railgun: streaming windows for mission critical systems
- Railgun: managing large streaming windows under MAD requirements
- Railgun: A new weapon for mission critical streaming tasks

In this summer internship, you will help us to further optimize Railgun to have even lower latencies, by implementing some of the optimizations that we already have in mind, but also identifying and proposing new ones.

You will join the Railgun team, participate in the team day-to-day ceremonies, and develop production-ready code that will also be reviewed by the team.

Requirements:

- Experience with Java.
- Knowledge about concurrent programming (race conditions, locks, threads, ...).
- Interest in investigating performance issues, and coming up with creative solutions to solve them.
- Basic knowledge of Git, Maven, and JUnit is a plus.

Internship plan (at least 8 weeks):

- Deep dive on Railgun (1 week).
- Write Railgun micro-benchmarks to get a baseline performance measure (2 weeks).
- Identify possible optimizations and implement them (4 weeks).
- Validate the optimizations (1 week).
- Ship them to production :p

Artificial Intelligence for Cloud Operations (AIOps)

Original proposal by: Ricardo Freitas & João Ascensão | Type: Cloud Engineering & Data Science | Preferential location: TBD

AIOps is a trending term that applies AI/ML capabilities to cloud operations in order to automate processes and detect or even predict anomalies through big data analysis and event correlation.
With Feedzai's constant growth, the scalability challenge is constantly in our mind. One of the current challenges is related to operational noise, for example with alert fatigue.

This summer internship proposal intends to create an AIOps service focused on detecting and preventing false alerts in our Feedzai Cloud environments.

Keywords: Go, Kubernetes, Observability, AI

Requirements:

- Programming skills (Go, Python or similar languages);
- Experience with Infrastructure-as-Code concepts;
- Basic experience with data science and machine learning;
- Basic experience with kubernetes and CI/CD tools;

The internship plan (total of 12 weeks) is:

- Understanding the problem with alerts fatigue (1 week);
- Exporting the relevant data to train predictive model (1 week);
- Data curation and exploratory data analysis (2 weeks);
- Model training, evaluation, and selection for the use case (2 weeks);
- Develop the AIOps service (3 weeks);
- Promote Service to production in dry-run mode and get insights (2 weeks);
- Present final results to relevant stakeholders (1 week);

Research - Machine Learning Research

Original proposal by: João Ascensão | Type: Data Science | Preferential location: Lisbon/Porto

The Research Data Science team ensures that Feedzai builds cutting-edge Artificial Intelligence to manage financial risk.

Some of the research topics we focus on, and which you may work on, include:

- **Better AI models for financial crime detection** (such as fraud, money laundering, or identity scams). Work in this area tends to focus on novel machine learning algorithms or techniques aimed at improving detection results;
- Automating and improving the **process of building successful models for real-world machine learning tasks** leveraging techniques such as Data-Centric AI and AutoML;

- Improving the effectiveness and the efficiency of real-world **machine learning systems in production**, including MLOps and AI Observability;
- Human-in-the-loop machine learning, namely **active learning** and annotation.

In this Summer Internship, you will be able to collaborate with Feedzai's Machine Learning Research team in refining a research problem and proposing and testing a solution. You will also learn about AI in the industry, while building your machine learning and research skills.

Apply if you are curious, eager to learn and looking for impactful research problems.

Requirements:

- Programming skills (preferably Python);
- Experience with collecting, analyzing, and exploring data and experimentation;
- Familiarity with data science and machine learning problems;
- Interest in learning more about data science, machine learning and applied research.

Responsibilities:

- Collaborate with the Feedzai Research team in refining a research problem;
- Learn and apply machine learning state-of-the-art;
- Design and execute data science experiments;
- Share your knowledge.

Figma chart component library

Original proposal by: Rita Costa, João Palmeiro | Type: Research Data Visualization | Preferential location: Lisbon

Data is at the core of Feedzai. Everyday, both ourselves and our users need to explore incredible amounts of data and communicate insights. Data visualization provides an effective way of communicating. To create the most useful charts, we design mockups of the visualizations and iterate with users. Figma is Feedzai's go-to design tool for that purpose. In this internship, you will be developing a library of chart components in Figma that leverage our design system. This will be used by the Data Visualization (DV) and UX teams when designing data visualizations in Figma. With this work, you will contribute to speed-up and improve DV and UX teams' development process.

Requirements

- Have a keen eye for visual aesthetics
- Experience with UI design tools (Figma, Sketch, Adobe XD)
- Experience in JavaScript and web development (nice to have)
- Experience in Data Visualization (nice to have)

Goals

- Define a set of chart elements and chart types to be included in the library
- Develop components that are in line with DV and Feedzai's design system
- Communicate and promote usage of the library within DV and UX teams

Internship plan

Week 1

- Understand Feedzai's work
- Understand how Figma is currently used for creating charts by DV

Week 2

- Select chart elements and chart types to be included in the library

Week 3 and 4

- Create MVP of the component library with a limited selection of chart types and elements

Week 5

- Extend charting library to include other chart types and elements

Week 6 and 7

- Option 1: Develop a Figma plugin similar to Datavizer with Feedzai's styles
- Option 2: Discovery on how to export Figma charts to React
- Option 3: Continue work on extending the charting library

Week 8

- Present the results by doing an internal presentation and writing a tech blog post on Feedzai's blog