Index

- Home

Hacking Materials User Interface

About

This project aims to provide a web-based user interface for a standard process in Matminer. Retrieving data from databases, user selection of features to be extracted within the databases, performing simple machine-learning tasks (scikit learn, Keras), and visualising results.

In computational materials engineering, engineering new materials are accelerated by avoiding expensive and lengthy experiments to demonstrate materials' performance. Data mining methods discover better materials by searching computer-generated databases with simulations predicting using high-throughput and high-performance computing. Unfortunately, the translation of this new thinking into engineering practice is still in its infancy with some frontrunners (e.g., Tesla). A barrier is materials engineers' software tools. An engineer with materials domain knowledge needs to access and process these data efficiently to make an informed decision for eventual machine learning strategies.

The Matminer python library provides a framework to simplify the process of data retrieval, feature extraction, machine learning and visualisation. But even a python tool is a high barrier in engineering practice.

Contact Details

Quick Links

Slack	Trello	GitHub
link	link	link

Client

Name	Email
Dr Christian Brandl	christian.brandl@unimelb.edu.au

Staff

Name	Email	Role
Mauro Mello Jr	mauro.mellojr@unimelb.edu.au	Supervisor
Eduardo Oliviera	eduardo.oliveira@unimelb.edu.au	Subject Coordinator

Team contacts: **BoxJelly**

Name	Email	Role	Student ID
Zhaoqi Wang	zhaoqiw@student.unimelb.edu.au	Team Rep & Frontend developer	1018092
Dara O hEidhin	doheidhin@student.unimelb.edu.au	Backend developer	1006587
Felipe Leefu Huang Li	fleefuhuangl@student.unimelb.edu.au	Frontend developer	1202652
Radhimas Djan	djanr@student.unimelb.edu.au	Backend developer	1146240
Yaoming Xuan	yaomingx@student.unimelb.edu.au	Backend developer	1167302

Team contacts: **Redback**

Name	Email	Role	Student ID
Alastair Daivis (Team rep)	adaivis@student.unimelb.edu.au	Team Rep	359 101
Chunbaixue Yang	chunbaixue@student.unimelb.edu.au		1208333
Ghina Yashar	gyashar@student.unimelb.edu.au		1274878
Mamta Lopes	mlopes@student.unimelb.edu.au		1157314
Sanjeevani Avasthi	savasthi@student.unimelb.edu.au		1101265

Team contacts: **Bluering**

Name	Email	Role	Student ID
Yanan Liu (Team rep)	yananl7@student.unimelb.edu.au	Team Rep	1289747
Hongpeil Lu	hongpeil@student.unimelb.edu.au		1275238
Jiahao Ju	jiahaoj1@student.unimelb.edu.au		1128182
Rui Zhang	rzzhan2@student.unimelb.edu.au		1221568
Xinle Yu	xinley@student.unimelb.edu.au		1294310

Recent space activity

Felipe Leefu Huang Lin

| Frontend workspace structure proposal updated 14 minutes ago • view change

Zhaoqi WANG

Sprint 2 - Checklist (retrieved from Mauro) updated 14 minutes ago • view change

Dara O hEidhin

Sprint 2 - Reflection updated 20 minutes ago • view change

Zhaoqi WANG

Sprint 2 - Reflection created 30 minutes ago

Dara O hEidhin

Sprint 2 - Checklist (retrieved from Mauro) updated 34 minutes ago • view change

Space contributors

- Felipe Leefu Huang Lin (14 minutes ago)
- Zhaoqi WANG (14 minutes ago)
- Dara O hEidhin (20 minutes ago)
- Mauro Mello Jr (2 days ago)
- Yaoming Xuan (2 days ago)
- ...

| Milestones

- Sprint 1 - Assessment Checklist
- Sprint 1 - Reflection
- Sprint 2 - Checklist (retrieved from Mauro)

Sprint 1 - Assessment Checklist

Sprint 1 Assessment Checklist

This checklist helps you double check our work for Sprint 1.

Background description, client goals, motivation

- [x] Project overview, background and goals were created.
- [x] DO-BE-FEEL list and GOAL MODEL were created.
- [x] The goal model is consistent with the client understanding of the problem and with DO-BE-FEEL list.

Personas

Make sure that your Personas satisfies the following criteria:

- [x] 2-3 personas were developed to help with requirements validation.
- [x] Personas are based on the research done by students and the discussion with industry partners.
- [x] Personas are inclusive and diverse.

Analysis of requirements (User Stories or Use Cases)

- [x] The analysis of requirements was performed on most of the existing requirements.
- [x] The [new set of] requirements is consistent to the scope of the project, completely cover the new capabilities required by the client and are well documented/structured/organized on Confluence.
- [x] The requirements can be documented in the form of user stories or use cases, supplementary specification of design/implementation/deployment requirements, prototypes, and others. It may also be necessary to be explicit about what is not in scope to define the scope boundary more clearly.

Development environment

- [x] Confluence is organized (cover page, project details, requirements, technical details about the project, meeting minutes and so on).
- [x] Trello (or Github projects or JIRA) is created, structured and organized.
- [x] README file is updated and provide details about the project, workflow (branches/naming conventions and so on).

Plan

- [ ] Approved 3-team colaboration plan approved by Eduardo
- [x] A plan (or discussion on what to do next) was provided (requirements to develop, technologies to use, infrastructure to deploy the project) for Sprint 2 and Sprint 3.
- [x] Requirements were estimated and prioritized.
- [x] Backlog items can be found in Trello (or Github project or JIRA).

Meetings

- [x] Meetings are recorded in Confluence and only. They were NOT exported to Github as they're part of internal process.

GitHub

- [x] Folders are structured (On Canvas, visit Assignment -> "Sprint 1: Confluence Space, project background and elicitation documents" page: you can find requirements for folders' structure.)
- [x] Sprint 1 documents were exported from Confluence and added to the repository (and are updated)
- [x] README file is updated and explain the team's repository
- [x] A baseline tag was generated for this Sprint (On Canvas, visit Assignment -> "Sprint 1: Confluence Space, project background and elicitation documents" page: you can find requirements for the baseline tag)

Additional Information

do you have any other additional information you'd like to share with us? Please add it here.

Sprint 1 - Reflection

Sprint retrospective (Reflection):

What went well:

During this sprint, what went well was how the team could adapt and act quickly on deciding how to collaborate within three groups. Throughout sprint 1, we decided to make a weekly meeting. Before the meeting, each group will do their respective task, and then during the meeting, they will work together on the task. By doing this, the team is able to work together and finishes the user stories, motivational model, persona, business case, and QA plan. At the end of the sprint, the group planned to distribute the work for the next sprint by spreading tasks using epic. Each group will be assigned to an epic to make it easier to split up the works. Based on our experience in this sprint, we are confident that in the next sprint, the three groups will be able to work as a team on our tasks respectively and finish them on time.

What did not go well:

- The main thing that did not go well in this sprint was the administrative task on the universities side. Our supervisor suggests us the three groups work together as a team. However, the universities are pretty slow to give access to the tools the team needs, such as GitHub and confluence access to each group. We are not able to work as efficiently due to this. Fortunately, for sprint 1, there hasn't been a development that requires a GitHub, so it isn't affecting the work on the team.

- Finding the common availability between 15 people for the meeting is difficult. We mitigate this by having team representatives on each team and communicating it between the team. These approaches work from what we experience, but it is quite a high effort for the team representative and creates an increased risk of information getting lost in transmission.

What to improve:

The main thing we need to improve on in the next sprint is the meeting with other groups because we found that scheduling meeting between 3 teams is sometimes tricky. Improving it is crucial because, for the next sprint, we planned to distribute two epic task for each group. However, there will be user stories that correlate with each other. Due to this, we will need to ensure that the member working on the user stories is in the meeting if it's associated with other group user stories.

What we think of the client interactions:

We feel that in sprint 1, Dr. Christian Bradl was very cooperative throughout the process. He always responds to our email, explains things clearly on the specification of the task, and explains what features are good to have or could have. He is also cooperative in setting up a meeting while considering the three teams' times.

Sprint 2 - Reflection

Sprint retrospective (Reflection):

What went well:

- Actively and rapidly learned and built confidence in new knowledge related to assigned tasks.
- Designed and applied Trello boards and cards well for everyone to easily track the current process of development.
- Reduced the number of team meetings, but used them more efficiently by preparing agendas / questions and prioritising topics.
- Many tasks were close to "done", which means we started to speed up.
- Defined a set of useful workflow and standards to make sure teams work smoothly in the development phase.

What did not go well:

- Amount of time for setting up workspace of each person during their different workplace experience.
- Code review took too much time to be complete and approved.

What to improve:

- Record meetings or tutorials when people get help from others.
- Limit deadlines for each code review.

Change log

Version date	Editor	Comment
19 Sep 2022	Zhaoqi WANG	Initial draft

| Development

| Development Process

- All tests are required to pass in CI before landing a pull request
- Sprint lifecycle:
    - Sprint Kickoff:
        - Review and re-estimate tasks: user stories get t-shirt size and priority
    - Development:
        - Feature kickoff:
            - Specify test cases and acceptance criteria
            - Tasks are estimated in number of days to complete using the magic estimation approach
        - Code reviews:
            - Require tests pass in CI before merging
            - At least one other RedBack member must approve the pull request before it can be merged
            - At least one BoxJelly member working on the same piece of technology must approve the pull request before it can be merged
            - At least one BlueRing member working on the same piece of technology must approve the pull request before it can be merged
            - All test cases and acceptance criteria identified in kickoff must be satisfied
        - Use auto-formatters to maintain code quality
        - Branching
            - Use the format `feature/t-<ticket>` as a feature branch template, where `<ticket>` is the Trello card number
            - `<username>/idea` for scratch / experimenting branches
            - `main` is the main branch
            - We will follow the following guidelines: https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow (we probably only need main, release, scratch, and feature branches)
    - Release management:
        - Deploy as required

** Co developed with other teams, written up by Redback.*

| Quality Assurance Guidelines

Work quality guidelines
- Code should be able to run, there's no syntax error
- If test(s) are included in the pull request, they should pass
- All previous tests should still pass with the new changes in the pull request
- A branch should not be put up for a pull request if it has merge conflicts with main, all conflicts should be resolved before that
- Code should be understandable and contain documentation
- Code should not include global state

Code review guidelines
- Has to be reviewed by 1 member from the other 2 teams
- Pull request should have assigned reviewers within 24h once it's no longer DRAFT and should be reviewed (either approve or require changes) within 48h
- Pull requests should include commit messages describing the work you've done and steps you did to verify your work
- Pull requests will be reviewed by using the Review Changes button under the Files changed tab
- Pull requests should only be merged/rebased by the creator of the pull request
- Branch that has been successfully merged/rebased to main should be deleted by the creator of the pull request
- Reviewers should follow the work quality guideline to review the code

Acceptance criteria definition guidelines
- It should be defined from the user's point of view
- It should contain a list of steps to test the desired functionality

Definition of done (for user story)
- Acceptance criteria should be defined for the user story and pass
- All related code has passed code review and merged to main

*Co-developed *with all other teams, written up by Redback*

| Frontend workspace structure proposal

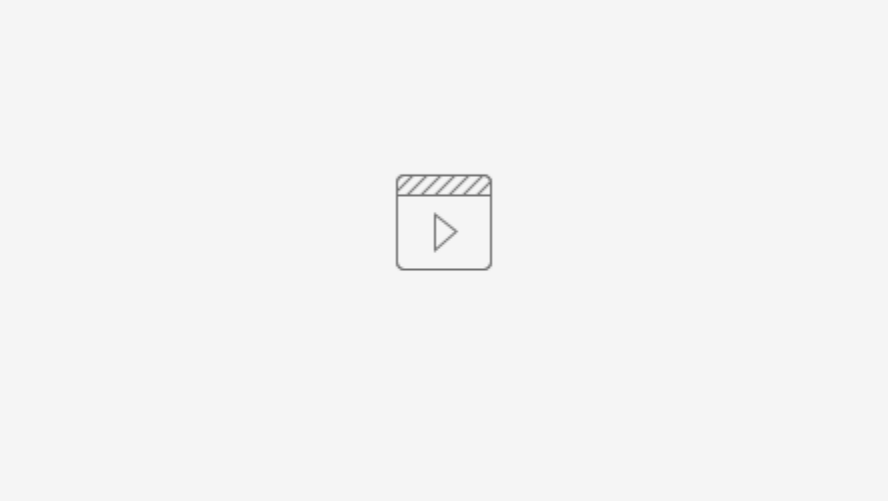
Proposed by	Ghina Yashar
Presented to	All frontend contributors from teams RedBack, BlueRing and BoxJelly
Proposal date	September 1 2022
Status	APPROVED - September 2 2022
Approvers	• Mamta Lopes(RedBack): 1/9/2022 • Felipe Lin (BoxJelly): 2/9/2022 • Rui Zhang (BlueRing): 2/9/2022

Proposed Structure

Please note: All the names used below can be replaced if needed, the focus of this proposal is more on the structure rather than the naming.

Summary video

If you don't like reading, please watch the video below for a quick overview of the proposed structure. The sample code snippets shown in the video are copied below as well.



Proposal summary.mov

Summary in writing

The structure I'm proposing would follow this rough directory tree:
Sample directory structure

```
|_assets
|_src
    |_components
        |_exampleComponent
            |_examples.tsx
            |_index.tsx
            |_test.tsx
            |_styled.tsx
        |_ dropdownSelectStepType
            |_examples.tsx
            |_index.tsx
            |_test.tsx
            |_styled.tsx
        ...
    |_steps
        |_datasetSelection // (e.g.)
            |_index.tsx
            |_test.tsx
            |_HelpModal
                |_index.tsx
                |_styled.tsx
                ...
            ...
        ...
    |_sections
        |_appHeader
            |_index.tsx
            ...
        |_appBody
            |_InputPanel
                |_index.tsx
                ...
            |_ViewingWinow
                |_index.tsx
                ...
            |_index.tsx
        |_appFooter // amendment suggested by Felipe
            ...
    |_App.tsx
    ...
|_package.json
|_README.md
...
```

The main ideas of this are as follows:

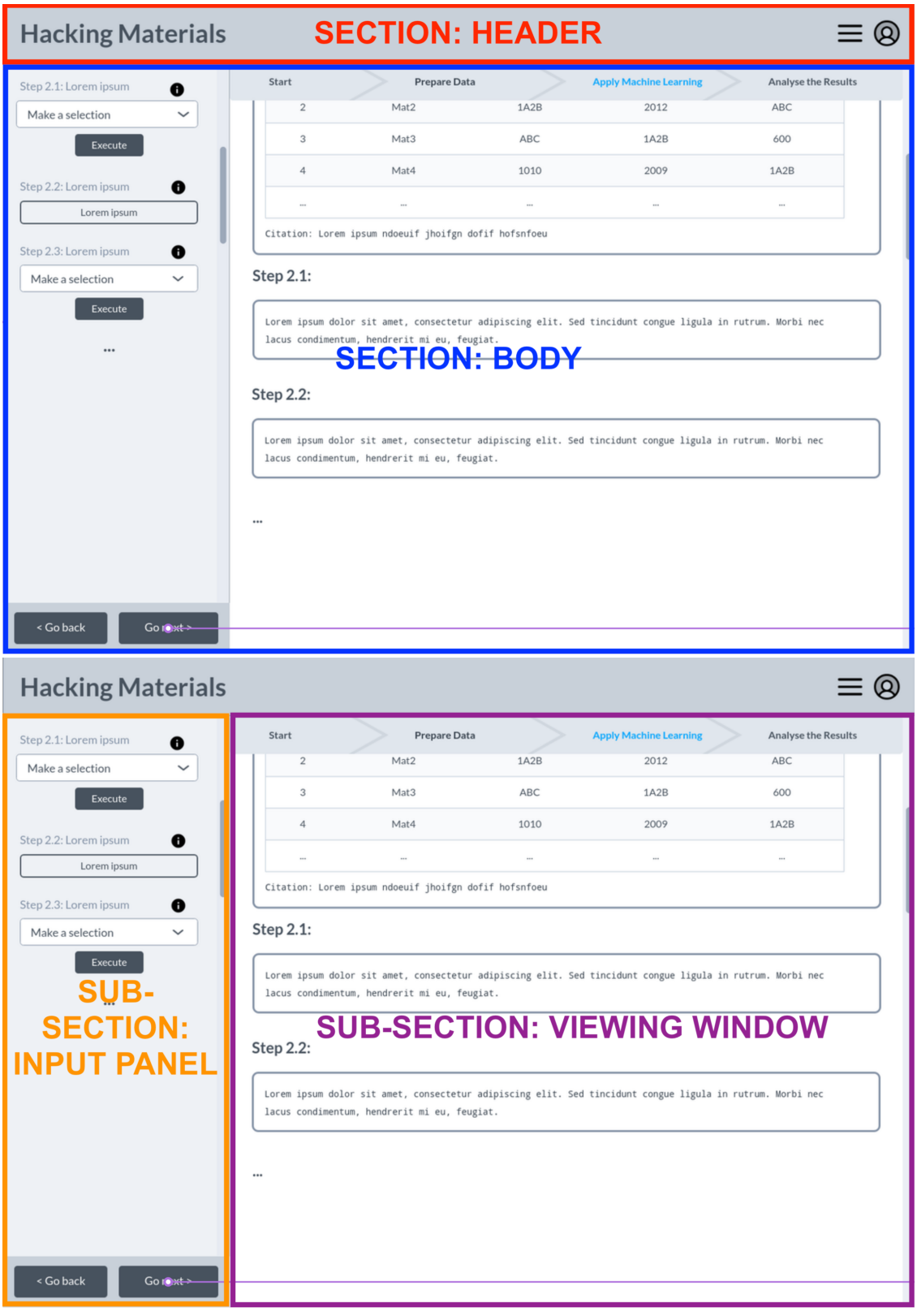
`Sections`

By referring to the low-fidelity prototype created earlier in the project, we divide the main application page into 2 main sections:

- Header: the top bar, which does not need to have context of what stage the user is up to and what's happening at any given point.
- Body: Includes 2 subsections that both need to know which stage the user is at (e.g. "Pre-process data" or "Apply machine learning"):
    - Left-side panel: named in the structure as `InputPanel`. Example code for this panel and how it shows the workflow steps is included in the sample code section below.
    - Main window on the right: named in the structure as `ViewingWindow`

AMENDMENT - 2 SEPTEMBER 2 2022

Felipe Lin (HA-BoxJelly) suggests potentially adding an `appFooter` as well. No objections to this so far.



`Components`

This folder will mainly contain all reusable components, e.g. Button, Tooltip, Modal, etc.

Notably, some of these reusable components would be "step types", e.g. `DropdownSelectStepType`. This example step type refers to the entire object shown below, including a step number, title, tooltip, dropdown list, button and whatever else may be needed. We would create this as a reusable component because many steps have similar requirements, e.g. selecting a dataset and selecting a featurizer should both be dropdown list type steps.

Steps

The word "steps" in this section refers specifically to the workflow steps that would be shown in the input panel, e.g. Dataset Selection step, Featurizer Selection step, etc.

A separate folder is created for these so that there would be a clear pattern that is easy to follow whenever more steps need to be added. Each step would use a step type component that is imported from the `/components` folder. E.g. the `DatasetSelectionStep` would use the `DropdownSelectStepType`, as shown in the sample code snippet below.

Sample code snippets

Sample src/steps/datasetSelection/index.tsx

```
import DropdownSelectStepType from '../../components/dropdownSelectStepType';
import HelpModal from './HelpModal';
...
const DatasetSelectionStep = (props) => {
    ...
    const STEP_KEY = "dataset_selection"

    const options = api_call_here() // calls backend API to get the dataset options

    const onSubmit = selected_value => send_to_backend() // send to backend using api

    return (
        <DropdownSelectStepType
            stepNumber={props.stepNumber}
            title="Select Dataset"
            description="bla bla"
            tooltipContent={HelpModal}
            options={options}
            onSubmit={onSubmit}
        />
    );
};
```

Sample src/sections/appBody/InputPanel/index.tsx

```
import DatasetSelectionStep from '../../../steps/datasetSelectionStep';
import FeatuirzerSelectionStep from '../../../steps/featuirzerSelectionStep';
...

const InputPanel = (props) => {
    ...
    const { stage } = props;

    if (stage === 1) {
        return (
            <div>
                <DatasetSelectionStep
                    stepNumber="1.1"
                    data={data}
                    handleChange={handleChange}
                />
                <FeatuirzerSelectionStep
                    stepNumber="1.1"
                    data={data}
                    handleChange={handleChange}
                />
                ...
            </div>
        );
    } else if (stage === 2) {
        return (
            <div>
                ...
            </div>
        );
    }
};
```

| Specifications

Links
| Project Description
| Motivational Model
| Personas
| User Stories
| Prototype
| Business Case
| Plan

| Product Description

Goals

- Help material engineers and people with interest in the field to predict the properties of a new material with simulations to avoid redundant experiments.
- The idea is reasonable but still in infancy in practice. We need to make it happen.

Sponsor

- Dr Brandl
    - Lecturer in UOM. Teaches computational material engineering.
    - Has some knowledge about python and Jupyter Notebook, but knows very little about programming and algorithm.
    - Very cooperative.
    - Regular meeting: 1:00 PM on Friday.

Scope - User Types

- Regular user
    - General user with knowledge and interest in materials engineering.
    - Has no or very little experience in python, programming or algorithms.
    - Would benefit from an easy-to-use application to assist in retrieving data and performing applied machine learning functionalities.
- Pro user
    - Regular user with access to additional features:
        - In control of the codes of the ML methods and the datasets.
            - Able to modify the python codes in the interface.
            - Able to introduce new datasets from new sources.
            - Able to add new features and delete existing features (the latter is optional).
        - Account management (not mentioned in the meeting but should have this function).
        - For now, the pro user is Dr Brandl himself.

Scope - Functionalities

- Web page
    - Able to demonstrate diagrams.
    - Able to show more details about data entries in the diagram on click.
    - Provide hints and guidance for new users (low priority).
    - Provide an interface for pro users to code directly.
    - Login through Unimelb emails
- Backend server
    - at least allow 30+ users to operate concurrently since there are 30 students in need of this software this semester.
    - Runs on unimelb cloud server (i.e. Spartan).
- Machine learning module
    - Feature engineering
        - Able to generate a report about the importance of every feature in a specific task.
        - Analyze the relationship between different features (optional).
        - Allow pro users to add new features.
        - Allow all users to define which features to use.
    - Algorithms
        - Provide various ML algorithm options (and their parameters perhaps) for users to choose from.
        - The logic can be exported into a jupyter notebook file (the nature of the exported file may change in the future).
        - Pro users should be able to modify the ML algorithms and even upload their own script.

Technologies

- Hosting
    - University of Melbourne Cloud Services
- Frontend
    - React
    - Typescript
- Backend
    - Python
    - Flask
    - Docker
- Admin & Collaboration Tools
    - Confluence
    - Trello
    - Github
    - Slack

Stakeholders

Stakeholder	Role	Interests and expectations project	Project Influence / Importance
Dr. Brandl	Main Customer	Receive a properly working application with at least the scope must have features	High
Students from teams BoxJelly, Bluering, Redback	Product Developers	Acquire knowledge and experience. Fulfil subject's goals and hurdles	High
Subject Supervisor	Project Supervisor	Guide students to follow the good practices of software project development	Medium
Future application users	Client	At the moment N/A	Low

Additional Details

- Dataset is clean and reliable. It shouldn't require much preprocessing.
- The project should be utilise the Matminer material analysis library.

Change log

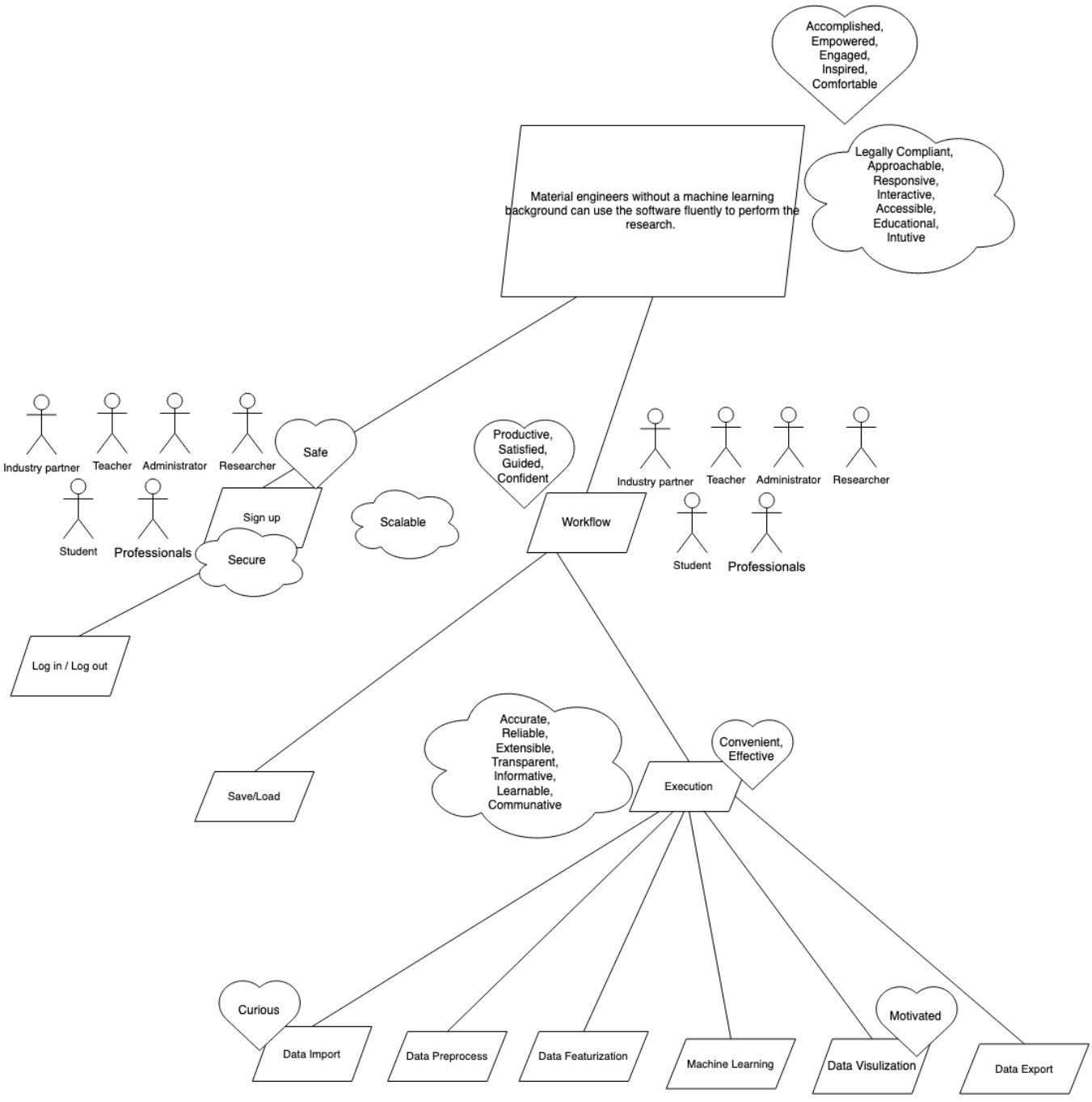
Date	Version	Author	Comment
15 Aug 2022	1.0	Felipe Leefu Huang Lin	First draft
19 Sep 2022	1.1	Dara O hEidhin	Changes based on Mauro's feedback

| Motivational Model

Do-Be-Feel List - Collaboration Version 1 (Sprint 1 - Final)

Who (users)	Do (functional goals)	Be (qualitative goals)	Feel (emotional goals)
Students	Add more database, machine learning method and plot types	Accessible	Accomplished
Administrators	Compare data using tables & plots	Accurate	Comfortable
Professionals	Data Pre-processing: Calculate descriptive statistics	Approachable	Confident
Industry Partners	Data Pre-processing: Consider anonymized data	Communicative	Convenient
Teachers	Data Pre-processing: Overview of the current import data	Educational	Curious
Researcher	Data Pre-processing: Reduces noise and eliminates ambiguity	Extensible	Effective
Code maintainers	Data Pre-processing: Standardizing data to bring it into the formatting range	Informative	Empowered
	Data Visualization: Data processing: Tabular data & Plotted Graph	Interactive	Engaged
	Edit python code directly in the interface	Intuitive	Guided
	Export input data	Learnable	Inspired
	Export jupyter notebook file	Legally Compliant	Motivated
	Export output data tables and figures	Reliable	Productive
	Featurization data: Add multiple composition-based features	Responsive	Safe
	Featurization data: Add multiple simple density features	Scalable	Satisfied
	Import Data: Create working spaces when importing	Secure	
	Import Data: Drag and drop import of files	Transparent (progress, error messages, notebook export...)	
	Import Data: Import data files (CSV, XES, Parquet) from local system		
	Log in/Log out		
	Machine Learning: Define input data and output data: Splitting data into training, test, and validation sets		
	Machine Learning: Determining model features and training the model: Configure and adjust hyper parameters for optimum performance		
	Machine Learning: Evaluate model performance and establish benchmarks: Continuous measurement and monitoring of model performance		
	Machine Learning: Evaluate model performance and establish benchmarks: Evaluate models using validation methods and validation datasets		
	Machine Learning: Get model results: The most important features of the current ML model		
	Machine Learning: Select the machine learning model to be used		
	Maintain software		
	save/load workflows		
	Sign up		

Goal Model - Collaboration Version 1 (Sprint 1 - Final)

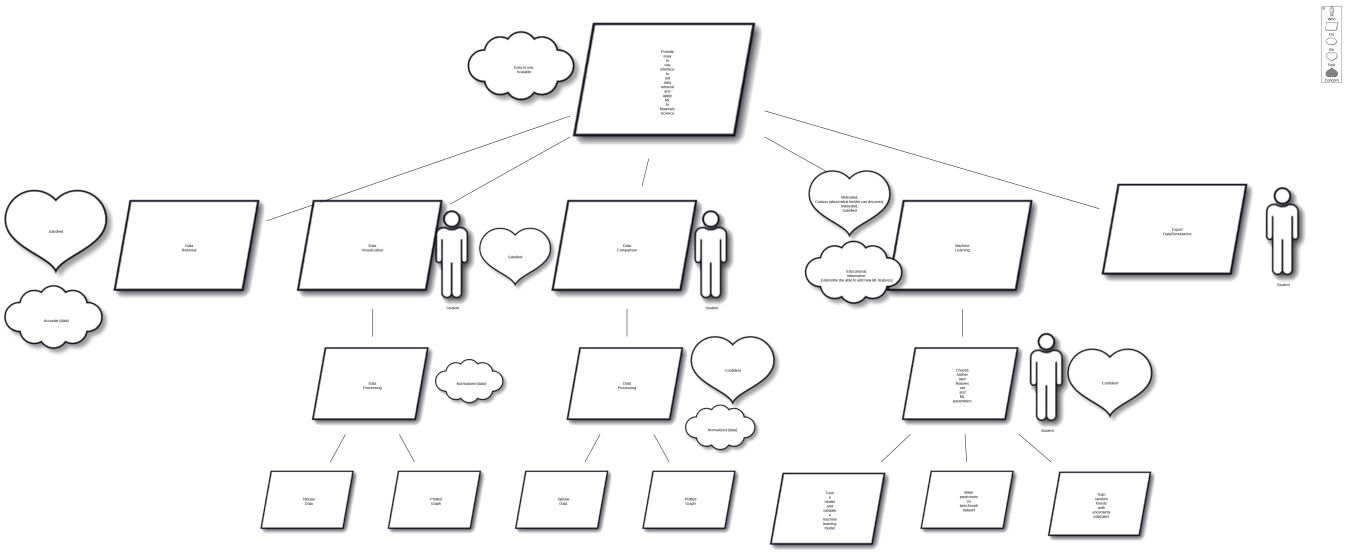


Do-Be-Feel List - BoxJelly Version 1

Who	Do (Functional Goal)			Be (Quality Goal)	Feel (Emotional Goal)
Pro user	Create new features based on existing features			Easy to use	Motivated
Pro user	Edit python code directly in the interface			Accurate (data)	Satisfied
Pro user	Adding more database, machine learning method and plot types			Reliable	Satisfied
Students	Data Retrieval			Normalized (data)	Interested
Students	Data Visualization	Data Processing	Tabular data	Educational	Confident

Students	Data Comparison		Plotted Graph	Informative	Curious (about what he/she can discover)
Students	Machine Learning	Choose his/her own features set and ML parameters	Train a model and validate a machine learning model	Extensible (be able to add new ML features)	Inspired / Satisfied
			Make predictions on benchmark dataset	Scalable	
			Train random forests with uncertainty estimates	Reliable	Confident
Students	Export Data/Simulations			Easy to use	Convenient
Students	Running environment			Stable	Satisfied
Students	Log in			Intuitive	Safe

Goal Model - BoxJelly Version 1



Change log

Version date	Editor	Comment
08 Aug 2022	Dara O hEidhin	Initial template
15 Aug 2022	Yaoming Xuan	Fill in the information about the students
17 Aug 2022	Dara O hEidhin	Merged do-be-feel and goal model into one page
17 Aug 2022	Felipe Leefu Huang Lin	Restructure Do-Be-Feel list
17 Aug 2022	Felipe Leefu Huang Lin	Upload first draft of goal model
18 Aug 2022	Zhaoqi WANG	Filled in some blanks
18 Aug 2022	Zhaoqi WANG	Added row on Do-Be-Feel list
18 Aug 2022	Dara O hEidhin	Added row on Do-Be-Feel list
18 Aug 2022	Radhimas Djan	Added row on Do-Be-Feel list
21 Aug 2022	Zhaoqi WANG	Updated with merged results (Final) with other teams
22 Aug 2022	Felipe Leefu Huang Lin	Added 3 teams collaboration Do-Be-Feel list and Goal Model version

| Personas

Persona 1 (Regular User) - Prepared by team BoxJelly

Assol Anahita

age: 22
residence: Melbourne
education: Material Engineering
occupation: Student
marital status: Single

"It's SO time consuming to do material researches and get decent results through just a semester."

Motivation: As a material engineering graduate student, Assol gets frustrated and demotivated when she can't make sense of the data she has because she doesn't have a tool or sufficient programming/machine learning skills to process the material data. She is also frustrated that she can't use machine learning algorithms to help her engineer new materials even though she is told by her supervisor that this idea works in theory.

Comfort With Technology

PROGRAMMING WITH PYTHON

MACHINE LEARNING

CLOUD BASED STORAGE

MATERIAL SCIENCE

Criteria For Success:

Assol can perform materials data requests/retrievals and accurate materials property predictions supported by Machine Learning technology with easy to follow steps button clicks user interface.

Needs

- Easy-to-use interface Material science data processing and retrieval application
- A tool to predict property of a material with assistance of Machine Learning technology without prior knowledge of Python and Machine Learning programming

Wants

- A data mining application that helps her researches
- A better understanding on how Machine Learning can help her to learn more about a material
- Ability to use ML algorithms as a black box
- Freedom to select features on her own terms
- A tool to accelerate research progress

Values

- Convenience
- Quickness
- Safety
- Understandable

Fears

- Spends hours without getting anything done because she neither has an adequate tool to do data mining, nor the programming skill to analyse the data herself
- Have to conduct countless experiments to figure out the properties of the materials
- Hard to choose suitable ML algorithms

Persona 2 (Regular User) - Prepared by team BlueRing

Gray Zhou

age: 28
residence: Ningde, Fujian, China
education: Master of Material Engineering
occupation: R & D Engineer of Polymer
marital status: Single

"It is fantastic to apply a multi-function online tool with ML methods if it is efficient and reliable. Nobody will refuse a tool that can save his time"

Motivation: Gray Zhou is a R & D Engineer of polymer in a battery factory. His work is searching for better materials for battery production. Gray spends lots of time testing different materials, but some of tests are waste of time because of the poor performance observed. He needs a system that can predict some useful properties of materials so that he can remove samples with low predicted performance and boost the research. His company provides some solutions, but they are awkward and only have limited functions.

Comfort With Technology

PROGRAMMING WITH PYTHON

MACHINE LEARNING

CLOUD BASED STORAGE

MATERIAL SCIENCE

Criteria For Success:

Provide a website or online-tool with quick, visual interface which can help him in daily development of new materials.
A successful product should help him save noticeable time on data processing and provide reliable prediction of properties.

Needs

- Retrieve and extract required data, process the data with ML methods to get some properties
- Provide graphs which can be modified with interface about predicted properties
- Help finding the material with best predicted properties

Wants

- Ability to interact with the graph to further compare serval materials in detail
- Upload data from his lab for predicting
- Explain what ML method the system applied and how it helps the prediction
- Continue his work on mobile devices without gaps of interaction

Values

- Easy to get started on both desktop and mobile
- Efficient back-end process
- Abilities to select functions and filter results
- Well organized visualization of interface and graphs

Fears

- Not enough guidance in the web or tool so him may feel confused to find functions he wants.
- Lackingunderstandofwhatthesystemdone,then reducing the confidence level of his report
- Frequently unavailability

Persona 3 (Pro User) - Prepared by team RedBack

Alex

age: 45
residence: Melbourne
education: Masters Degree in Physics
occupation: Materials Engineer
marital status: Divorced without kids

"There has got to be a better way to do this."

Motivation : As an experienced Materials Engineer, Alex's job requires him to narrow down candidate materials by performing physical experiments to choose a material which can takes years to do. He needs a tool that can speed up the process by narrowing down candidate materials for experimentation using Machine Learning and simulations.

Comfort With Technology

PROGRAMMING WITH PYTHON

MACHINE LEARNING

CLOUD BASED STORAGE

MATERIAL SCIENCE

Criteria For Success:

Alex can find the right materials efficiently, with accurate results and that matches the client's requirements.

Needs

- Products to accelerate his workflow
- Access to wide variety of related tools and resources

Wants

- Suitable models and featurizers for different use cases
- Demonstrate reproducible results to his clients
- Share resources with others
- Refining generated workflow to reuse

Values

- Extensibility
- Accuracy
- Reliability
- Responsiveness
- Scalability
- Transparency

Fears

- Tool is too inflexible
- Losing access to progress on his work
- Not being able to verify his results
- Not having support with the tool

Change log

Date	Version	Author	Comment	File
15 Aug 2022	1.1	Felipe Leefu Huang Lin	First draft	persona.pdf
16 Aug 2022	1.2	Yaoming Xuan	version 2 of the persona	persona.pdf
16 Aug 2022	1.3	Felipe Leefu Huang Lin	Used name generator to create a random name	student.pdf
16 Aug 2022	1.4	Yaoming Xuan	Converted to docx file	persona....960.docx

16 Aug 2022	1.5	@Yaoming Xuan	small modifications	
17 Aug 2022	1.6	Zhaoqi WANG	Suggestions on Quote, Needs, Wants, Values, Fears and Logo (Brand). Issues: 1. can't upload more picture 2. extended to 2 pages 3. Name seems to be automatically changed from time to time	
20 Aug 2022	1.7	Yaoming Xuan	Generate a pdf file with latest content	
22 Aug 2022	1.8	Zhaoqi WANG	small modifications	
22 Aug 2022		Felipe Leefu Huang Lin	Added personas created by teams BlueRing and RedBack	
22 Aug 2022		Dara O hEidhin	Embedded a pngs of our personas for site PDF export compatibility	
18 Sep 2022	1.9	Felipe Leefu Huang Lin	Revised persona following spring 1 feedback.	
18 Sep 2022		Felipe Leefu Huang Lin	Updated newest version of team Redback and Bluering persona	

Link to make additional changes: https://personagenerator.com/7300af72-1c96-11ed-8d77-d742c7efee51/7300af73-1c96-11ed-8d77-5f5915d1141f

| User Stories

Prioritization Technique

We used the MoSCoW prioritization classification.
Must have - must be included in the scope of the project, we defined this all the must have user stories can create a MVP
Should Have - should be included in the scope of the project
Could Have - could be included in the scope of the project
Won't Have - will not be included in the scope of the project

Teams Collaboration - Version 1.2 (Sprint 1 - Final)

ID		Role		Action	Epic		Goal	Size (days)	Priority	Assigned Team
30	As a	general user	I want to	be able to view the citations for used featurizers	Input Data	so that	I could be know more about the source of the featurizer (legally compliant)	1	1 - Must have	RedBack
32	As a	general user	I want to	browse and select built-in featurizers	Input Data	so that	I can discover ways of manipulating my data	1	1 - Must have	RedBack
34	As a	general user	I want to	browse built-in datasets	Input Data	so that	I can discover data to experiment with	1	1 - Must have	RedBack
19	As a	student	I want to	quickly browse the Materials available in the database for retrieval and simulations	Input Data	so that	I can quickly perform queries.	3	1 - Must have	RedBack
21	As a	general user	I want to	be able to select datasets from exisiting databases	Input Data	so that	I do not have to worry about how the data is loaded	3	1 - Must have	RedBack
37	As a	general user	I want to	be able to preview the input data	Input Data	so that	I could explore the data	1	2 - Should have	RedBack
25	As a	general user	I want to	Select specific features from a dataset	Input Data	so that	I can improve the precision of my model	3	2 - Should have	RedBack
13	As a	Pro user	I want to	add new features	Input Data	so that	they can be reused in the future	5	2 - Should have	RedBack
28	As a	general user	I want to	be able to reference / view citation for original data sources	Input Data	so that	I can retrieve data.	1	3 - Could have	RedBack
18	As a	pro user	I want to	be able to apply new featurizers	Input Data	so that	I can create new features	3	3 - Could have	RedBack
1	As a	student	I want to	clean and tune data input	Input Data	so that	I have less noise on visualizations.	5	3 - Could have	RedBack
29	As a	student	I want to	save project specific data/checkpoints	Administration	so that	I can pick up where I left off for specific projects	1	1 - Must have	BoxJelly
35	As a	pro user	I want to	export model selections, parameters, and data flows	Administration	so that	I can save my work and share it with others	1	1 - Must have	BoxJelly
36	As a	pro user	I want to	import exported model selections, parameters, and data flows	Administration	so that	I can continue work I had previously saved	1	1 - Must have	BoxJelly
20	As a	student	I want to	Create an account using single-sign on, restricted to the *.unimelb.edu.au domain	Administration	so that	my research remains private	3	1 - Must have	BoxJelly
23	As a	pro user	I want to	Control job execution	Administration	so that	I can start, view progress of, and cancel jobs related to my project	3	1 - Must have	BoxJelly
10	As a	pro user	I want to	be able to opt in to pro-user features	Administration	so that	I can access pro user features	5	1 - Must have	BoxJelly
38	As a	pro user	I want to	have my pro user settings persist on each visit	Administration	so that	I don't have to reconfigure settings to use the features I need	1	2 - Should have	BoxJelly
24	As a	student	I want to	recieve provided hints and guidance for new users	Administration	so that	I can quickly learn how to use software	3	2 - Should have	BoxJelly
14	As a	pro user	I want to	easily find and read documentation on the pro features	Administration	so that	I can use them with ease	5	2 - Should have	BoxJelly

17	As a	pro user	I want to	Be kept informed about job status	Administration	so that	I can avoid polling my workspace to check for results	3	3 - Could have	BoxJelly
6	As a	pro user	I want to	have access to more processing power	Administration	so that	I can run more complex operations or use more data	5	3 - Could have	BoxJelly
31	As a	general user	I want to	able to select a Machine Learning model	Machine Learning	so that	I could use it to train and run the data	1	1 - Must have	BlueRing
33	As a	general user	I want to	browse built-in ML models	Machine Learning	so that	I can discover ways of manipulating my data	1	1 - Must have	BlueRing
39	As a	user	I want to	be able to select split ratio of data	Machine Learning	so that	to train and test the model	1	2 - Should have	BlueRing
26	As a	pro user	I want to	have the option to change the hyperparameters used in the machine learning model	Machine Learning	so that	I can fine tune my test results.	3	2 - Should have	BlueRing
15	As a	pro user	I want to	be able use additional ML models	Machine Learning	so that	I can improve accuracy	5	2 - Should have	BlueRing
7	As a	pro user	I want to	combine multiple models together	Machine Learning	so that	I can model more complex data manipulations	5	3 - Could have	BlueRing
22	As a	general user	I want to	see clear annotation or explanation of data points and features	Data Visualisation	so that	I can understand the results of the analysis	3	1 - Must have	TBD - after the completion of all assigned "must have"s
8	As a	student	I want to	use different type of plotting graphs	Data Visualisation	so that	I have flexibility to visualize data according to my needs.	5	1 - Must have	TBD - after the completion of all assigned "must have"s
9	As a	general user	I want to	able to view and plot the results of the model	Data Visualisation	so that	I could analysis and visualise the effects of the model	5	1 - Must have	TBD - after the completion of all assigned "must have"s
12	As a	student	I want to	export my work to a Jupyter Notebook	Jupyter Notebook	so that	I can extend my work beyond the capability of the application	5	2 - Should have	TBD - after the completion of all assigned "should have"s
2	As a	general user	I want to	attach comments to workflow objects	Jupyter Notebook	so that	I can document my work	5	3 - Could have	TBD - after the completion of all assigned "could have"s
4	As a	Pro user	I want to	edit python code on the interface	Jupyter Notebook	so that	I can have control how the ML algorithms works	5	3 - Could have	TBD - after the completion of all assigned "could have"s
5	As a	Pro user	I want to	upload my own script (in python) if possible	Jupyter Notebook	so that	I can extend the tool to support custom models and featurizers	5	3 - Could have	TBD - after the completion of all assigned "could have"s
27	As a	pro user	I want to	be able to access new databases	External Data	so that	I can access addional data	3	2 - Should have	TBD - after the completion of all assigned "should have"s
3	As a	Pro user	I want to	be able to add new datasets in the future	External Data	so that	if there's a new dataset that can be used on a new project, it can be added instantly	5	3 - Could have	TBD - after the completion of all assigned "could have"s
11	As a	student	I want to	analyze the relationship between different features		so that	I can identify which features I need to select for my analysis	5	2 - Should have	TBD - after the completion of all assigned "should have"s
16	As a	general user	I want to	add specific materials to the workflow for analysis		so that	compare the performance of the specific material my client or I choose with other material	3	3 - Could have	TBD - after the completion of all assigned "could have"s

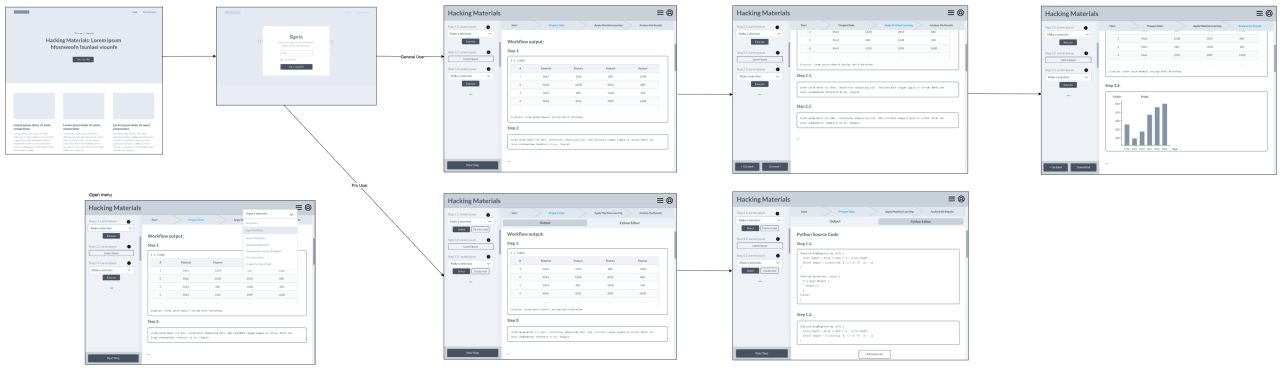
Change log

Version date	Editor	Comment
08 Aug 2022	Dara O hEidhin	Initial template
17 Aug 2022	Felipe Leefu Huang Lin	Added first user stories
18 Aug 2022	Dara O hEidhin	Entered user stories in table and added some more

18 Aug 2022	Zhaoqi WANG	Added more user stories
18 Aug 2022	Yaoming Xuan	Added pro user stories
18 Aug 2022	Radhimas Djan	Added more user stories and edited some of the user stories
19 Aug 2022	Radhimas Djan	Added merged user stories
21 Aug 2022	Zhaoqi WANG	Version 3 user stories
22 Aug 2022	Felipe Leefu Huang Lin	Added the final version of collaboration user stories.
22 Aug 2022	Zhaoqi WANG	Added "assigned team" column
18 Sep 2022	Zhaoqi WANG	Implemented changes according to the feedback
18 Sep 2022	Dara O hEidhin	Deleted old versions of user stories table and only kept current table and changelogs

| Prototype

Teams Collaboration - version 1 (Sprint 1 - Final)



Descriptive Notes (recorded by team redback)

- Landing page:
  - Static page with information about the app and product
  - Link to access the app
    - On click, it opens a login modal
    - Once user is logged in, they're redirected to the app
- Single page app:
  - Top bar:
    - User profile button at the top opens a menu to give the user the option to log out
    - Menu button at the top has options to import or save a workflow, download it in different formats, start over, a link to the documentation and a toggle to enable the pro view.
  - General user:
    - The workflow is divided into major and minor steps. Each major step would have its own page. User can go back and forth between the major steps as needed.
    - Left panel:
      - All the minor steps are numbers and named to guide the user
      - Inputs can be of different types
      - Each step has a tooltip button that would open a modal with guidance information about the step
      - The steps and options in the left panel should always be the same no matter what selections the user made in previous steps. Any step that requires customised inputs would open in a modal.
        - Example 1: Step 3.1 might be "Selecting a plot type". As there is a known, limited list of different plot types, this step may be a drop-down menu that is displayed directly in the left panel.
        - Example 2: Step 3.2 might be customising the selected plot's configuration options. As different plot types may need different configuration options, these options will not be displayed in the panel directly. Instead, the panel will include only a button that says "Configure plot", which would open a modal with the specific options applicable to the selected plot type.
      - Pinned buttons at the bottom of the panel: navigate between the different major steps. The last step page may also have a button to download the full workflow.
    - Viewing window:
      - At the top of the viewing window, the user can see the progression of major steps with the current step highlighted.
      - The output of each minor step is labelled with the step number and contained inside a box. The output inside the box is the same output produced by running the python code, simply copied over for transparency.
      - The outputs from the previous pages are also always displayed, so it's not just the outputs of the current page.
      - Where a resource with citations is used, the citations will be automatically printed after the output of the step where the resource was selected.
  - Pro user:
    - Left panel: has all the same options as a general user, plus additional buttons to configure their own settings as needed
    - Viewing window: the window has 2 tabs:
      - Output: same the as the viewing window of the general user
      - Python source code:
        - An editable view of all the code generated by their selections, looks similar to a Jupyter notebook.
        - User can add new cells as desired
        - Brings up the following question: what happens if the user edits the code generated by one of the steps? This may lead to inconsistencies between what is shown in the step's input field and what the code now actually does. This is an implementation decision so is not a major concern right now, but one option that we decided to show in the prototype is that the step's input in the left panel would change to say "Custom" or something similar, indicating that the configuration was changed.

Change log

Version date	Editor	Comment
20 Aug 2022	Felipe Leefu Huang Lin	First prototype draft made in collaboration with team redback and bluering

| Business Case for Cross-team Collaboration

Project Hacking Materials Cross-Team Collaboration Model Proposal

COMP90082 Software Project SM2 2022

The current situation

We currently have three teams working with the same client, Dr Christian Brandl, on the project Hacking Materials (“HA”). The client expressed that he would not be interested in three different versions of the same product, and would prefer we work together to be able to build one more complete final product.

Additionally, it is clear from our discussions with the client that any resulting single product will be difficult to decompose into totally independent components. This means that the deliverables produced by each team will be dependent on those produced by other teams, requiring collaboration on design, development, and project management tasks.

This issue is compounded by the fact that each team has a separate workspace set up by the university, in particular the separate GitHub repositories. This makes it extremely difficult to share resources including source code and development resources, and to determine how much progress is being made by each team, and on which tasks.

Our biggest concern is that our teams will be assessed separately. Working on the same product makes us inherently reliant on each other, which could affect the assessment of each team. We understand that this complicates the situation, and have come up with a working agreement that we believe would allow us to deliver the product that the client wants while minimizing the risks to the assessable components of the project.

Obstacles

We have identified several issues with the way the project is currently being approached, including the following:

- Project scope is larger than what could realistically be accomplished by one student team.
- If the teams were to work completely separately on different parts of the product, it will be difficult to organize ourselves in a way that ensures the needs of all teams are met.
- Finding common availability between 15 people for meetings is difficult. As such, cross-team communication so far has been done mainly through team representatives. This approach is slow, high effort for team representatives (as they need to be aware of everything their teams are doing at all times) and creates an increased risk of information getting lost in transmission.

Proposed work structure / way of working

The structure we are proposing is as follow:

- The teams would work on the same GitHub repository.
- Each team works as a full-stack team focusing on a particular epic of stories,
- Someone from each team is involved in each technical area of the product (Backend/ Frontend/ Machine Learning).
- The members of the different teams who are within the same technical area cooperate to ensure they follow the same standards and processes. This will help spread some of the cross-team communication load away from the team representatives.
- Each team will need to review the other teams’ work to ensure that it doesn’t interfere or affect their own work.
- Ownership of epics will be allocated to teams, but the teams will work together to ensure all high priority user stories are delivered first. This creates fewer dependencies between teams and allows each team to deliver fully functional components without relying on the others.
- Each team will be responsible for their own Confluence space, but the teams will maintain a similar structure to make navigation easy. Major structural changes will be reported and shared with the other teams during cross-team meetings.
- The teams will share certain deliverables within their Confluence spaces so that the client would not have three different versions of the same document.
- Sprint documents exported from each team’s Confluence space will be included within their own folder in the shared GitHub repository.
- The teams will take turns organizing meetings with the client, supervisor and each other.

Requested actions

- **Decision from teaching staff:** University administration hasn't made a decision on assessment criteria, or whether the teams can collaborate on this project. This prevents us from adopting a shared working model because we are not sure whether our project submissions will be accepted by the university, or whether we will be otherwise penalized for taking this approach. This was discussed with our supervisor, Mauro, and it was agreed that a response would need to be available by Tuesday 3pm.
- **Shared GitHub repository:** If our proposal is accepted, we would need a shared GitHub repository so that Dr. Brandl will not have three copies of each deliverable.

Change Log

Date	Version	Author	Comment
15 Aug 2022	1.0	Redback Team	First draft
19 Sep 2022	1.1	Dara O hEidhin	Changes based on Mauro's feedback

| Plan

Options:

Scenario 1 - If the projected is divided between the three teams then we expect to split each epic equally between the three teams.

Scenario 2 - If we work as separate teams the user stories will be addressed in order of priority and entered into Trello. Please refer to Business Case and User Stories

Collaborative changes, on GitHub, will pass a reviewing process with each sprint being released with a tag.

Result:

It was agreed to follow Scenario 1

Change Log

Date	Version	Author	Comment
15 Aug 2022	1.0	Redback Team	First draft
19 Sep 2022	1.1	Dara O hEidhin	Changes based on Mauro's feedback

| Acceptance Criteria

ACCEPTANCE CRITERIA

USER STORY ID	USER STORY	GIVEN	WHEN	THEN
20	Create User Database and Endpoints	I have a registered account for use of this software	I update any relevant personal information	That updated information persists in the database and is linked to my account
			I make changes to my workflow	That workflow data can be saved in the database and is linked to my account
			I delete my account	All linked data is deleted from the database
20	Create an account using single-sign on, restricted to the *.unimelb.edu.au domain	I own a Unimelb account and would like to access the web application features	I click in the login button	I can sign-in using my unimelb.edu.au email as a form authentication
		I am a first time user and don't own a Unimelb account	I click in the login button	If I try to sign-in using a non-unimelb email I receive a notification that only unimelb.edu.au emails are allowed
29	Save project specific data/checkpoints	I am logged into my account	I make changes to my workflow	That data can be saved to the database under my user account
35	Export model selections, parameters, and data flows	I am logged into my account	I click on the export button	I can download a file that has all the data needed to recreate my current workflow state
36	Import exported model selections, parameters, and data flows	I am logged into my account	I click on the import button	I can upload a previously exported workflow state
		I have uploaded a previously exported workflow state	The file passes all data integrity checks	My UI will match the state represented in the imported file
10	opt in to pro-user features	I am a pro user and would like to access my pro-user features	I click in the opt button	I jump in a new window with all pro-user features accessible
23	Control job execution	I have a task in progress	I click in the start, view and cancel buttons	The running tasks can be started, viewed and canceled

Change Log

Date	Version	Author	Comment
17 Sep 2022	1.0	Felipe Leefu Huang Lin	First draft
19 Sep 2022	1.1	Dara O hEidhin	Added user stories from Sprint 2

| Sprint Artefacts

Links
- Sprint Review
- Retrospectives
- Product planning

Trello Board link

https://trello.com/b/PzZuNQMk/boxjelly-sprint

| Sprint Review

SPRINT NUMBER	LOCATION
1	Team Sprint 1 Review
2	To be done following delivery of Sprint 2

| Retrospectives

Sprint retrospective (Reflection):

Sprint	Link
Sprint 1	Sprint 1 - Retrospective

Team reps retrospective

What went well	Comments	What could be better	Comments	What is confusing	Comments
Collaborative design groups	People were more willing to talk in focussed discussions, and the meetings were much more productive	It's taking a long time for code reviews to start		Confluence PDF export is awful	Each team to ask if anyone is willing to spend time investigating ways to improve this, then on Thursday we'll assign someone
RedBack preparing tutorials and knowledge sharing		It's hard to tell what other teams are working on	Regular standups will help. Also shared trello boards.	Presentation requirements still not defined (not mentioned in lectures or on assignments LMS page)	Mauro will tell us something by tuesday
Ghina suggested creating sub-teams in Slack	Communication is clearer because slack groups are more focused, and we can mention specific teams instead of @everyone	We didn't do a reflection or retrospective (especially as a whole team)	This is now part of our regular (weekly) meetings	Teaching staff were not prepared for multiple teams collaborating	
Shared GitHub repository	Good infrastructure for future collaboration, easier to merge shared code	No regular standup meeting	Schedule 2/week (one for standup, one for client meeting planning)		
Visibility of other team's confluence pages	We can work out what other teams are working on, and their working situation. Makes sharing artefacts and documents easier.	Didn't prepare a way to track everybody's work	Shared trello board will help		
People asking for help / knowledge sharing (Dara, Zhaoqi)	Makes it easy to help others learn and then help them help other people learn	Not clear which parts of the project team members don't understand	Encourage people to ask questions, wait		
		BlueRing and BoxJelly teams confused about what they should do next and what other teams are doing	Covered sprint kickoff and development cycle, regular standups will help		
		Yanan not included in backend slack channel until Friday	It's hard for people to know what needs doing unless they're in the right slack channel		

| Product planning

| Resources

Team HA - Skills assessment

Email Address	Team	Matminer	React	Flask	Pandas	Scikit-learn	Numpy	Matplotlib	Seaborn	Plotly	Bokeh	Tensorflow	Keras	Pytorch	NodeJS	Pure HTML/CSS	Angular	Vue	Django	NodeJS	Any other suggestions for backend
Hongpei Lu	Bluering	0	0	0	2	3	2	0	0	0	0	0	0	0	0	1	0	0	0	0	
Jiahao Ju	Bluering	0	0	0	4	4	4	4	0	0	0	1	0	2	0	3	0	0	0	0	
Xinle Yu	Bluering	0	0	0	4	4	4	4	4	4	2	3	3	3	1	1	0	0	0	1	
Rui ZHANG	Bluering	1	2	3	4	5	4	0	0	1	0	1	5	3	2	1	0	0	0	0	Server/Cloud ope
Yanan Liu	Bluering	0	0	0	4	4	4	4	2	2	0	3	3	3	0	0	0	0	0	0	
Dara O hEidhin	Boxjelly	0	2	1	5	5	5	3	3	2	0	4	4	1	2	2	1	0	0	2	
Yaoming Xuan	Boxjelly	0	0	0	4	5	5	0	0	0	0	1	1	5	0	3	0	0	0	0	
Felipe Leefu Huang Lin	Boxjelly	0	4	2	1	1	1	1	1	0	0	0	0	0	4	4	3	0	3	3	
Zhaoqi WANG	Boxjelly	0	1	0	4	2	4	3	2	2	0	3	0	2	0	2	0	0	1	1	
Radhimas Djan	Boxjelly	0	0	3	4	4	4	3	3	1	1	3	3	3	2	2	0	0	3	0	
Mamta Ritesh Lopes	Redback	0	3	0	3	5	5	4	0	0	0	1	1	0	0	4	0	0	0	0	
Ghina Yashar	Redback	1	5	0	4	4	4	4	0	4	0	4	4	4	2	5	2	1	0	2	
Chunbaixue Yang	Redback	1	0	0	3	4	3	2	2	0	0	3	3	1	0	1	0	0	0	0	ASP.NET
Sanjeevani Avasthi	Redback	2	1	1	5	5	5	4	2	4	0	3	3	4	0	3	0	1	0	0	
Alastair Daivis	Redback	1	1	2	2	2	3	2	2	2	0	1	1	3	2	2	1	2	1	2	

Description	Files
Summary	Team HA - Skills ...- Team Skills.pdf
Distribution of skills	Team HA - Skills Assessment.pdf