Degree Pathway App

Group members:
- Connor Hoskin   2348624
- Alex Hopgood   2938001
- Lachy Kitchen   1281963

Our Chosen Datasets:
1. University of Otago's paper information.
   > This will be used in a variety of ways. A list of papers will be shown for users to select which papers they're enrolled in. Papers that students are enrolled in will be checked against the requirements for the student's major.
2. Degree subjects/majors.
   > Students will be able to select the major/majors and minor/minors that they are enrolled in.
3. Degrees.
   > Students will be able to select the undergraduate degree they are enrolled in.
4. *Stretch: Students Academic Record*
   > Depending on available development time, students should be able to upload their academic record to be automatically scanned instead of having to manually enter each detail of their study.

Why did we choose them?
1. By obtaining all of the data for papers registered within the University of Otago's (UoO) database, students can select papers they have taken/want to take, adding it to their total points completed.
2. By obtaining all of the data for degrees and their associated subjects and majors registered within the UoO's database, our app can ensure students meet their degree requirements including prerequisites and restrictions. Additionally, as a stretch goal, we would like our app to:
   a. identify the possible pathways students can take given their completed papers.
   b. allow students to upload their academic record, identifying the papers they've taken and validating if they meet the criteria for their major/majors and their degree.

What we are going to build:

- A full-stack multi-platform application that:
  - **Shall**
    - Read in our chosen data sources populating them into a form
    - Allow students to select papers they have/will take
    - Calculate and display completed/forecasted points their selected papers accumulate to
    - Enforce degree and paper requirements (prerequisites/restrictions and number of points required for each level e.g., 54 points at 300-Level)
    - Calculate students GPA
      - Include a menu (potentially a dropdown or sidebar option), acting as a grade calculator for each paper including a default rubric for the course assessments
        - Including weightings and marks for each assessment (e.g., Final Exam: 50%)
      - Analytics section to breakdown GPA across different subjects
      - Forcast students future GPA grade based on a students' prior results
  - **Should**
    - Identify possible degree pathways based on paper selection
  - **Stretch**
    - Allow student to upload and read in academic records
    - Populate their degree pathway based on their academic record

How we are managing our project

Jira will be used for project management.

We have chosen to utilise Jira as our preferred project management tool for our full stack app development for several reasons. Firstly, Jira's robust features, particularly the agile boards, offer an intuitive and flexible way to visualise and manage our tasks, user stories, and sprints effectively. The boards allow us to track the progress of individual tasks and user stories, providing transparency and real-time updates on the status of the project. Additionally, Gantt charts in Jira enable us to create detailed timelines, establish dependencies, and allocate resources efficiently, ensuring smoother coordination between team members and enhancing overall project planning. With Jira's extensive customisation options and integration capabilities, we can tailor the tool to fit our specific project needs seamlessly.

Overall, the combination of Jira's agile boards and Gantt charts will empower our team to maintain an organised and structured approach to project management, leading to increased productivity and successful delivery of our full stack app.

https://kitla230.atlassian.net/jira/software/projects/EA/boards/1

What technologies are we going to use to build this thing? (Subject to change)

- **Languages:**
  - C++
- **Frameworks:**
  - Flutter (Cross-platform UI SDK)
  - Google Test (C++ testing framework)
- **DevOps**
  - Git (Version control system)
  - Github (Code hosting platform)
  - Jira (Project management)
  - Jenkins (CI)
- **Tools**
  - Doxygen (Documentation)
  - Visual Studio Code (IDE)

Gantt Chart
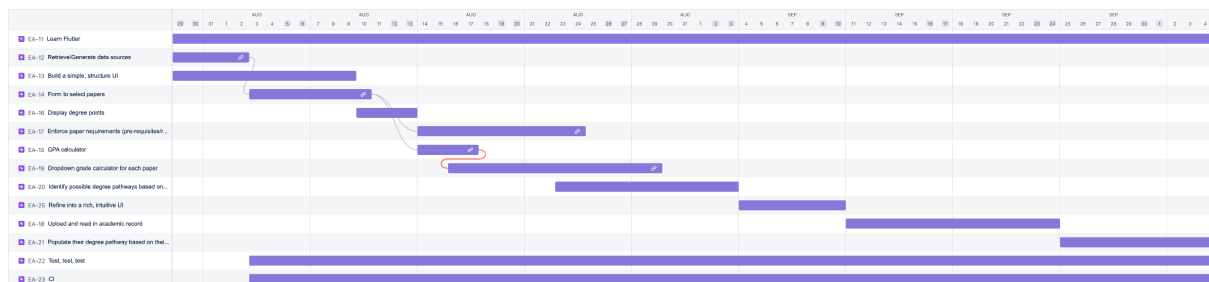
Risk Management

- **Risk identification**
  - Team exercise often through brainstorming
    - Engage in a set of activities that detect, describe and catalogue all potential risks to assets and processes that could have negatively impact business outcomes in terms of performance, quality, damage, loss or reputation.
- **Risk analysis**
  - Examining how project outcomes and objectives might change due to the impact of the risk event
  - For each risk
    - Judge probability of occurring (0-10 sale, 0-low, 10-high)
    - Judge cost of recovery (0-10 scale, 0-negligible, 10-catastrophic)
- **Risk planning**
  - Considers our responses or treatment of risks
    - Four risk response strategies are common
      - Do nothing (but plan)

- ○ Accept risk, but setup plans to deal with risks should they occur.
  - ● Risk tasking
    - ○ Create a task (as part of the schedule) that is meant to deal with the risk (e.g., task could be aimed at risk avoidance or transfer).
  - ● Risk tagging
    - ○ Mark specific risky tasks for implementation using specialised Agile techniques (e.g., pair programming or onsite customer). Raise awareness
  - ● Task dropping
    - ○ Remove the task from the project (task/risk may thus be avoided).
- ○ Plan for avoidance
- ○ Plan for impact minimization
- ○ Have contingency plans
- **Risk monitoring**
  - ○ Regularly re-assess each risk

Risk Assessment Schedule

Risk No	Risk	Likelihood	Impact	Risk Value/Exposure	Risk Reponse	Notes
1	Different development levels among the team causing delays in delivery	5	4	20	Risk Tasking	Individuals with higher development skills in certain areas can teach and supervise those with less proficiency
2	Team members operating on different time schedules (holidays etc)	3	2	6	Risk Tasking	Hold meetings to plan out tasks well ahead of time and ensure that everyone is aware of the same deadlines
3	Data we are provided for import may be corrupted/not exist	1	8	8	Risk Tasking	We will make our own data
4	First time working in an AGILE work environment, can lack structure/direction when combined with unclear goal.	2	3	6	Risk Tasking	Ensure everyone is familiar with the material taught, and advice given in lectures and labs as these tools will help give us direction and provide frameworks to help follow a clear direction and/or goal
5	Mechanical failure (laptop breaking)	2	6	12	Do Nothing (but plan)	Should this happen other group members will still have their own machines and the university has machines we can use to work on while the laptop is getting fixed/replaced
6	Any software technology we are using becomes unsupported or a paid service	4	8	32	Do Nothing (but plan)	We can plan to use alternative services in case the ones we are using become unavailable
7	Technical debt (implied cost of future reworking)	7	5	35	Risk Tasking	Prioritizing and addressing potential risks and challenges in the development process before they accumulate and create long-term technical debt
8	Using software that contains vulnerabilities	2	9	18	Risk Tasking	Do research on all existing software we use to look for bugs and ensure that it is legitimate
9	Team member sickness	5	4	20	Do nothing but plan	If a team member is sick other members will have spare time to help out on the sick members tasks
10	Disagreement over developement	2	2	4	Risk Tasking	If team conflicts accour, refer back to course coordinatitors for assistance
11	Lack of experience with technologies needed for systems development (e.g., Flutter)	10	4	40	Risk Tasking	Activily learn technology throughout implementation phase via online tutorials, StackOverflow, ChatGPT, etc.
13	Database of our choice is not supported by the Uni server (which could make CI is difficult to implement)	2	5	10	Task Dropping	Migrate to a different BDMS or find alternative ways of implementing quality assurance
14	Git merge issues	8	3	24	Risk Tasking	If merge conflicts occur, we create a task to handle these conflicts and sort them out.
15	Natural Disasters/Pandemic	1	7	7	Risk Tasking	Develop contingency plans, allocate resources, implement risk mitigation measures, establish communication plan, collaborate with relevant authorities, and regularly monitor and review to
16	Team Member Leaving Unexpectedly	1	8	8	Risk Tasking	Knowledge sharing and documentation, cross-training and skill development, recruitment and onboarding planning, workload reallocation, communication and support, and potential risk transfer

Similar existing products:

The University of Otago features a GPA calculator on its website where all grades and their associated values (9 for A+, 8 for A etc) are entered manually. While functional, this requires far more user input than our proposed app which would only require users to upload their academic record. It also makes no sense to make users manually input the values for their grades which are set values that don't change. Simply storing the values relevant to the grade without requiring user input makes much more sense.

At the time of writing the closest service to the proposed app that we've identified is the AskOtago service itself, where students can go to check if they meet the requirements to

graduate their degree, among other things. While this service is good, it would be beneficial if some of its features could be replicated in a single app, cutting down on both meetings with AskOtago and also workload for AskOtago staff.

- University APP, grades. (gpa calculator, not persistent, trying to remove)

Another existing product with similarities to our proposal is "Grades - Grade Calculator, GPA" (henceforth referred to as Grades) by Plutonium Apps, LLC. Available on the Apple App Store for iOS devices, Grades breaks students' enrollment down into terms, classes, and syllabus items. In each term, users can specify the class' name, grading type, how many credits the paper awards, and add individual syllabus items with their own grade percentages. Users can select a target grade, enter the grades they achieve for each syllabus item, and the app will calculate what scores are needed to meet the users target grade.

Grades also features a GPA calculator, where users can choose to either select the grades they received themselves or have the app calculate GPA based on students' syllabus item results. While Grades has a diverse array of features, the app has its drawbacks.

While Grades is free to download, many features are locked behind in-app purchases. These include adding additional terms past the default singular term, adding "grouped" syllabus items, extra credit syllabus items, and more.

Customer Interest:

Anecdotally, the members of this group have asked multiple fellow University of Otago students who have all said that they thought our app would be useful to them if it existed. As University of Otago students are the primary stakeholder for this proposal (aside from COSC345 faculty), this seems like good enough evidence that there is at least some customer interest for the app.