Agile Software Projects

November 2020

Contents

List of Figures

1 Aims and Objectives

1.1 Product Goals & Value Proposition

Set clear goals and concise and appropriate challenges which are measurable. Aims should be specific, with your objectives building up a bigger picture of what you hope to do. Goals and operations should be clearly specified.

1.2 Usability Goals

The theme of usability is key.

1.3 Measuring Success

This section should set out the measuring criteria for your work and your presentation will be judged in relation to these aims and objectives.

2 Market Analysis

Our project does not have a direct competitor. There are some utilities which partially solve the problem we are tackling - namely, tracking grades - but none of these enable the user to compare themselves to their peers, nor aggregates grade data. As such, we must cast a wide net to analyze our place in the market and consider the *alternative solutions*. We have identified these utilities which overlap with our project as well as tools which solve analogous problems in different domains in order to get a more comprehensive picture of where our product stands in the market.

2.1 Overview of Alternative Solutions

Our solution offers two mains functions: tracking grades and sharing/comparing grades. For each feature, we have documented the alternative solutions and their specific functionality and notable design choices.

2.1.1 Grade Tracking

1. Peer's Python command line grade calculator

2. Peer's Google Sheets Degree Planner

3. gpacalculator.io

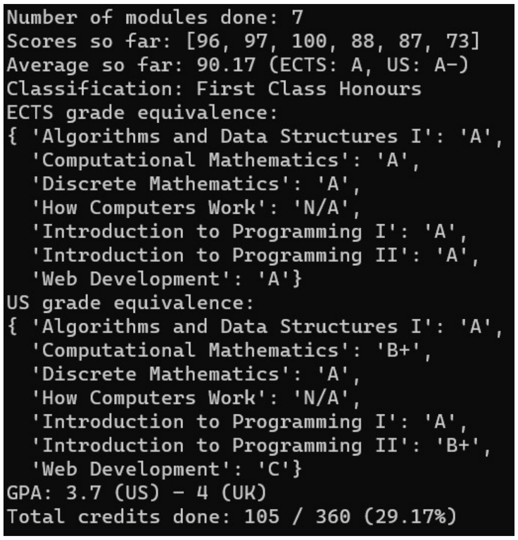
4. University of London Portal

The peer-created utilities and University of London portal were discovered through our teammates' participation in this course. gpacalculator.io is an example of many similar utilities that were found by searching 'grade tracking' and 'grade calculator' in Google.

1. Peer's Python command line grade calculator

`https://github.com/sglavoie/grades_calculator`

Users record their grades in a JSON file and run the script from the command line to see grade and other relevant information



```
Number of modules done: 7
Scores so far: [96, 97, 100, 88, 87, 73]
Average so far: 90.17 (ECTS: A, US: A-)
Classification: First Class Honours
ECTS grade equivalence:
{ 'Algorithms and Data Structures I': 'A',
  'Computational Mathematics': 'A',
  'Discrete Mathematics': 'A',
  'How Computers Work': 'N/A',
  'Introduction to Programming I': 'A',
  'Introduction to Programming II': 'A',
  'Web Development': 'A'}
US grade equivalence:
{ 'Algorithms and Data Structures I': 'A',
  'Computational Mathematics': 'B+',
  'Discrete Mathematics': 'A',
  'How Computers Work': 'N/A',
  'Introduction to Programming I': 'A',
  'Introduction to Programming II': 'B+',
  'Web Development': 'C'}
GPA: 3.7 (US) - 4 (UK)
Total credits done: 105 / 360 (29.17%)
```

Figure 1: Python calculator example output

Features:

- Track course and cumulative grades
- Converts to ECTS (Europe) and US scale
- Converts to US / UK GPA
- Determines which classification your grades fall into
- Tracks total credits done out of 360

Design Choices:

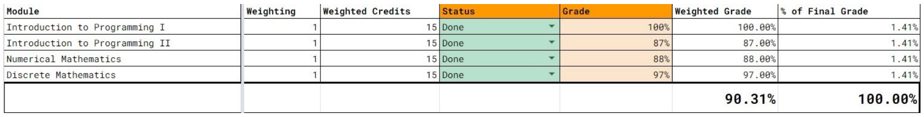
- Command line utility with input coming from JSON

2. Peer's Google Sheets Degree Planner

`https://docs.google.com/spreadsheets/d/1w5mFDaEB86q9zj-sdn_FZ8ivu5ALSfPsvBw8iOZ7qkY/edit#gid=0`

Users records their grades in a Google sheet to track their credits and grades.

Features:



Module	Weighting	Weighted Credits	Status	Grade	Weighted Grade	% of Final Grade
Introduction to Programming I	1	15	Done	100%	100.00%	1.41%
Introduction to Programming II	1	15	Done	87%	87.00%	1.41%
Numerical Mathematics	1	15	Done	88%	88.00%	1.41%
Discrete Mathematics	1	15	Done	97%	97.00%	1.41%
					90.31%	100.00%

Figure 2: Example excerpt from Google Sheets Degree Planner

- Track course and cumulative grades
- Tracks total credits done out of 360
- Shows which modules are optional

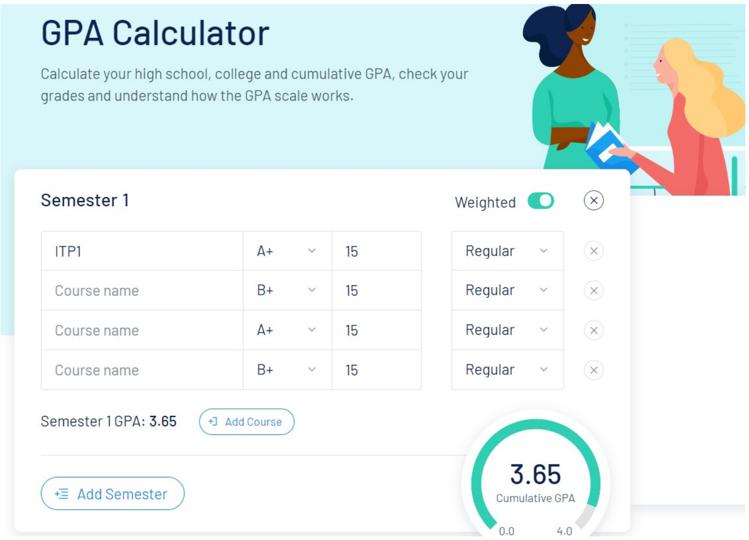
Design Choices:

- implemented on google sheets
- easy to share link

3. gpacalculator.io

`https://gpacalculator.io`

Users enters there grades in web form and sees cumulative results.



GPA Calculator

Calculate your high school, college and cumulative GPA, check your grades and understand how the GPA scale works.

Semester 1

Weighted

ITP1	A+	15	Regular
Course name	B+	15	Regular
Course name	A+	15	Regular
Course name	B+	15	Regular

Semester 1 GPA: 3.65

3.65 Cumulative GPA

Figure 3: Prominent landing page form on gpacalculator.io

Features:

- Calculate semester or total cumulative GPA from individual courses
- Offers different calculators for high school and college
- General information about GPA, grades, and different honors

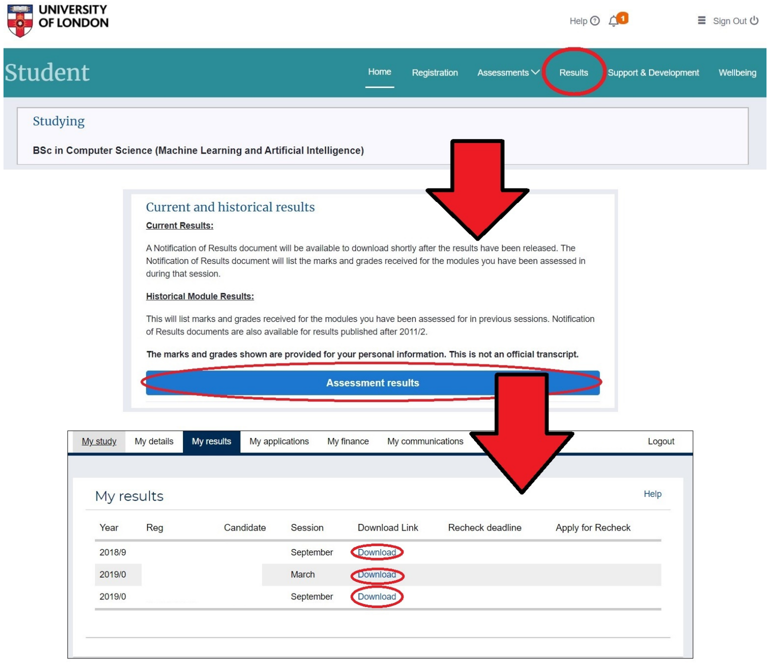
Design Choices:

- Input form is prominently placed. Immediately confronted with it when visiting the page.
- No login or form of persistence
- US specific

4. University of London Portal

`https://my.london.ac.uk/group/student`

Users can view grades from each semester, broken down by source (coursework vs. exam).



Studying

BSc in Computer Science (Machine Learning and Artificial Intelligence)

Current and historical results

Current Results:

A Notification of Results document will be available to download shortly after the results have been released. The Notification of Results document will list the marks and grades received for the modules you have been assessed in during that session.

Historical Module Results:

This will list marks and grades received for the modules you have been assessed for in previous sessions. Notification of Results documents are also available for results published after 2011/2.

The marks and grades shown are provided for your personal information. This is not an official transcript.

My results

Year	Reg	Candidate	Session	Download Link	Recheck deadline	Apply for Recheck
2018/9			September	Download		
2019/0			March	Download		
2019/0			September	Download		

Figure 4: Path to find grade download links on University portal

Features:

- Shows grade results for each semester

- Granular - shows grades for coursework and exam portion separately

Design Choices:

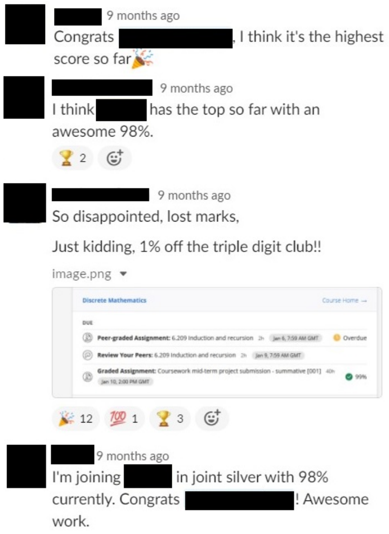
- Buried deep in menus
- No centralized place to view all grades nor cumulative grades
- download required for all but most recent results

2.1.2 Grade Sharing

1. University of London Slack workspace
2. glassdoor.com

We searched Google for 'grade sharing,' 'grade comparing,' 'compare grades to other students,' and other derivatives but found no relevant tools. Glassdoor was suggested as an analogous tool during a brainstorming session and the slack workspace is known to us through our participation in the degree program.

1. University of London Slack workspace
`https://app.slack.com/client/TDT1N1BUG/C01A9AR0A4C`



[redacted] [redacted] 9 months ago
Congrats [redacted], I think it's the highest score so far🎉

[redacted] [redacted] 9 months ago
I think [redacted] has the top so far with an awesome 98%.
🏆 2

[redacted] [redacted] 9 months ago
So disappointed, lost marks,

Just kidding, 1% off the triple digit club!!

image.png ▾

Discrete Mathematics   Course Home →

DUE

Peer-graded Assignment: 6.209 Induction and recursion 2h Jan 6, 7:59 AM GMT Overdue

Review Your Peers: 6.209 Induction and recursion 2h Jan 9, 7:59 AM GMT

Graded Assignment: Coursework mid-term project submission - summative [001] 40h Jan 10, 2:00 PM GMT ✓ 99%

🎉 12 💯 1 🏆 3

[redacted] [redacted] 9 months ago
I'm joining [redacted] in joint silver with 98% currently. Congrats [redacted]! Awesome work.

Figure 5: Students sharing midterm results and comparing to each other in course slack channel

This is the slack space open to all University of London online CS students. There are channels for each course. Grades are sometimes shared in threads after they are released.

Features:

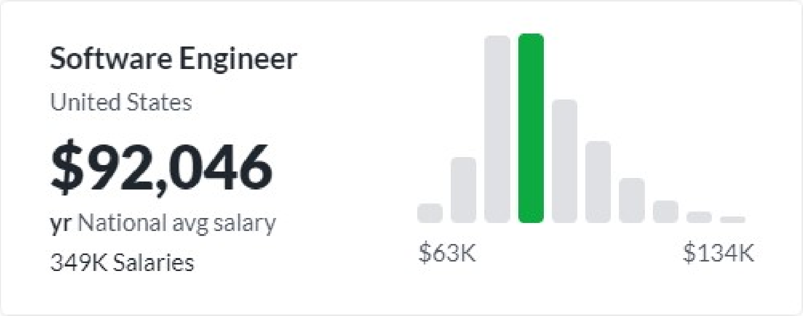
- Impromptu sharing
- Irregular
- Not solicited

Design Choices:

- Reports tend to be from people with high grades, or people with low grades who suspect an error with their results

2. glassdoor.com
`https://glassdoor.com`

A review and reporting website for employers. Collects textual reviews and salary data from users. Salary reports are analogous to grade reports.



Software Engineer

United States

$92,046

yr National avg salary

349K Salaries

Figure 6: Example of data and graph shown on glassdoor

Features:

- anonymously collects company reviews and salary information (including job title, locations of position, years of experience, etc.)
- job postings
- company profiles with generic info (number of employees, revenue, etc.)
- collects interview questions
- shows aggregate salary information for job titles (average salary for software engineer in a country, bar chart showing distribution of salaries)

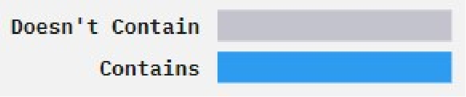
Design Choices:

- Must submit a salary to view salary information. Otherwise, salary collection form is not that prominent. The link to add salary can be found under ”salary” tab but it’s not highlighted in any way.
- Users define their job title - sometimes there are lots of close titles but it’s unclear if they are the same.

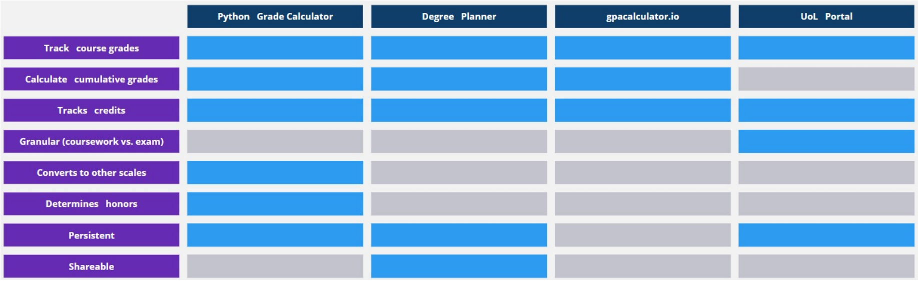
2.2 Comparing Alternative Solutions

The following tables compare the different features available across alternative solutions.

Legend

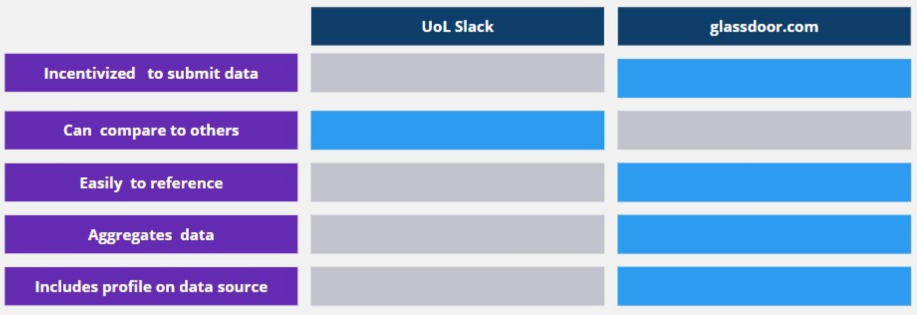


Doesn't Contain	⬜
Contains	🟦



	Python Grade Calculator	Degree Planner	gpacalculator.io	UoL Portal
Track course grades	🟦	🟦	🟦	🟦
Calculate cumulative grades	🟦	🟦	🟦	⬜
Tracks credits	🟦	🟦	🟦	🟦
Granular (coursework vs. exam)	⬜	⬜	⬜	🟦
Converts to other scales	🟦	⬜	⬜	⬜
Determines honors	🟦	⬜	⬜	⬜
Persistent	🟦	🟦	⬜	🟦
Shareable	⬜	🟦	⬜	⬜

Figure 7: Grade Tracking Comparison Table



	UoL Slack	glassdoor.com
Incentivized to submit data	⬜	🟦
Can compare to others	🟦	⬜
Easily to reference	⬜	🟦
Aggregates data	⬜	🟦
Includes profile on data source	⬜	🟦

Figure 8: Grade Sharing Comparison Table

2.3 Our Solution's Place in the Market

SWOT Analysis

2.3.1 Strengths

- Knowing you have a relatively high grade could be motivating / encouraging / validating
- Validation if your grade was low but many people report similar grades
- Highlighting of potentially more challenging courses for future cohorts to allocate more focus and preparation on
- Highlighting when many people consider their grades unjustifiably low which may help in organizing a more detailed, thoughtful request to the institution to review the grades or rubric
- Motivation to maintain ones standing at or above a certain percentile
- Leaderboard aspect "gameifies" grades and gives incentive to submit grades

2.3.2 Weaknesses

- Data reliability. We have no good way to validate data. We must disincentivize dishonesty and not assume 100% accuracy.
- A small sample size of participants will limit usefulness of averages
- Those with poor marks or marks they feel are unjustified will likely be less willing to share
- Those who fail or drop out may be unable (depending on integration) or unwilling to participate, which will artificially inflate all grade averages
- Without a representative sample of students, the averaged data, graphs, etc will be skewed and could lead to reduced acceptance or usage of the tool
- If we report anonymously, do we also store anonymously? If so then do we provide for the ability to remove data on request?
- If the tool is not completely opt-in / crowd-sourced then someone needs to moderate the data insertions and deletions.

2.3.3 Opportunities

- This project has few real competitors. There are some student created utilities but grade sharing/comparing is outside the scope of those utilities.
- The program size will continue to grow for a long time. We haven't reached the maximum number of cohorts and the recent cohorts have grown. The need exists, there just wasn't a reason to make it until now.
- If we are the first to make this, we can capture the interested user-base which should be a large moat.
- Develop this as with Slack integration so it can naturally link to the community of students that already exists, leveraging the University's policing of that population to being actual students
- Add some flags for people to rank their grades. Perhaps they felt the grade was unjustified, or that it was boosted or dragged down by group participants or life events (ie, only receiving a 50% in a course while feeling stress from COVID)
- Provide a field for students to provide short advice to others for each course they provide a grade for
- Store the user's grades for visibility of all their courses at once, which if public could be summarized quickly for sharing to others
- Provide an overall average per course, per semester, etc to help future students know what to expect and perhaps how much they need to prepare
- Provide for grade conversions using various methodologies to non-UK standards

2.3.4 Threats

- Time. Other student utilities show recognition of value for this type of product. Probably only a matter of time until someone else extends the grade idea in a similar fashion.
- Leak or theft of personally identifiable grade and contact information
- People could poison the data with dishonest grade reporting
- If the tool is completely opt-in / crowd-sourced ...
  - ... and we allow the removal of information then perhaps people would remove other people's data
  - ... people may impersonate other people along with reporting their real or falsified grades
- Loss of integrations / changes to any APIs we're using
- Data loss - backups

3 Planning and Requirements Gathering

3.1 Methodology

You should focus on how you hope to plan and gather requirements. We discussed some different methods in the lectures and discussion group activities. It is important that you evaluate the different methods in making your decisions and provide some written analysis in your reports, with clear critique and some functional understanding for the higher marks.

3.2 High-Level Requirements Funnel

3.3 Implementation Approach

4 Formative Testing and Evaluation

4.1 Identifying Users

This is the part of the project which requires the most research and understanding. For middle marks, you should provide a good, clear overview of techniques to identify and sample users, with a set of contextually relevant information and a balanced overall argument. This should bridge cohesively with your requirements engineering techniques, as well as your general research in target demographics, types of systems and processes involved

4.2 Testing & Evaluation Methods

Higher marks would require rigorous tests that are insightful and utilise a wide range of metrics to analyse success through different lenses.

4.3

5 Prototyping Techniques

You should describe your prototype, where strengths in different techniques lie and where they are used. This section should also detail the movement between low- fidelity and high-fidelity techniques, with a view to building the system and the types of technology involved. There should be evidence of iterative design and evaluation steps.

5.1 Techniques

5.2 Low-Fidelity Prototypes

5.3 High-Fidelity Prototypes

6 Evaluation Techniques

6.1 Critical Success Factors

Where do the critical success factors lie and what novel techniques could you use to measure your work?

6.2 Measurement of Success & Failure

How do you intend to measure success and failure in context?

6.3 Results

What works well about the system and where are the fundamental flaws?