**Project Plan Document**

**PDF Grader**

**Version 1.0**

**Table of Contents**

[**Business Requirements**](#_3znysh7) **1**

[Background](#_2et92p0) 1

[Business Opportunity](#_tyjcwt) 1

[Business Objectives and Success Criteria](#_3dy6vkm) 1

[Customer or Market Needs](#_1t3h5sf) 1

[Business Risks](#_4d34og8) 2

[**Vision of the Solution**](#_2s8eyo1) **3**

[Product Idea Statement](#_17dp8vu) 3

[Vision Statement](#_3rdcrjn) 3

[Major Features](#_26in1rg) 3

[Assumptions and Dependencies](#_lnxbz9) 4

[**Scope and Limitations**](#_35nkun2) **4**

[Scope of Initial Release](#_1ksv4uv) 4

[Scope of Subsequent Releases](#_44sinio) 4

[Limitations and Exclusions](#_2jxsxqh) 4

[**Business Context**](#_z337ya) **4**

[Stakeholder Profiles](#_3j2qqm3) 4

[Operating Environment](#_1y810tw) 4

[**Scheduling**](#_9f4g9jh05ojz) **5**

[Gantt Chart](#_ue0m9b6jgf8u) 5

[Key Milestones](#_b7bfvg1j3nb9) 5

[Resource Assignments](#_nig6ohd61l3z) 5

[Individual Responsibilities](#_4czh8sn6bfid) 5

[**Deliverables**](#_cqzqxtd9gksp) **6**

[Software/Hardware](#_9dirvameqi2j) 6

[Documentation](#_vgohaqb38usr) 6

[Key Presentations](#_rtp2pinwtto3) 6

**Revision History**

| **Name** | **Date** | **Changes** | **Version** |
| --- | --- | --- | --- |
|  | 1/21/22 | Initial Document | 1.0 |
|  |  |  |  |

# **Business Requirements**

## **Background**

*Oftentimes professors and their TA’s get their test grades out much after the test was taken. This is because grading tests can be a long process with little organization. This piece of software aims to remedy that by allowing professors to organize how they grade their tests in an efficient manner.*

## **Business Opportunity**

*The market opportunity exists within the fact that grading tests can be a painstaking process. This software is not revolutionary, but it will be effective. This means there could be competitors that make similar products with extra features, and those could compete with this product in a business setting.*

## **Business Objectives and Success Criteria**

*As far as the software goes, success will be measured by its functionality and ease of use. From a business standpoint, the overhead to keep it running is very low, as it’s really only the cost to make it. There is no server to upkeep and no domain to maintain. Because of this, any revenue that comes in from it will be profit, besides the time it will need to make updates. A strong SDD will allow us to achieve this objective. As far as a timeline goes, our software will be released by the end of our project time, and the steps taken will be anchored on consistent progression throughout the development that is planned in advance by our SDD’s.*

## **Customer or Market Needs**

*This product is meant to ease the difficulties and tedium of grading PDF assignments for teachers and TA’s, and this product is also meant to provide assignment grades to the students. Teachers and TA’s face the problem of grading assignments under a certain amount of time and that timeframe isn’t very feasible to accomplish when grading a PDF assignment with their current resources. This product fixes this problem by providing an interface that allows the teacher and TA’s to view questions and answers more easily and efficiently. Students face the problem of receiving the grade for an assignment and they don’t fully understand the reason behind their grades. This product fixes this problem by allowing the teacher to give a PDF to the student that gives specific details regarding the received grade.*

## **Business Risks**

*As far as clients are concerned, there are no other products that offer the same functionality as this product. However, there are some risks with this product.*

*In terms of buyer power there is very little risk since this is a free product, therefore no profit to lose. If this were a product that would yield a profit, then the risk would be if the buyer stopped using the product either due to inefficiency or substitution. This can be resolved by ensuring the product functions efficiently and monitoring any competition as necessary.*

*In terms of supplier power, the main risk that exists is if for some reason the suppliers that provide needed resources for this product weren’t able to provide these resources anymore. This can be resolved by looking into what resources are needed for this product and analyzing the probability of such scenarios happening. This can also be resolved by ensuring a degree of success for any businesses that would provide resources needed for this product.*

*In terms of threat of new entries, there is also little risk due to few products having the same functionality as this one. However, this scenario can happen once the product is available to customers since competition can observe this product’s functionality and implement it in a new product. This can be resolved by updating the product regularly and potentially adding new functions over time, that way competition will consistently lack the functionality this product offers.*

*In terms of threat of substitution, there is also little risk due to few products having the same functionality as this one. There are other products that can assist in PDF grading, but they offer different functionality. However, substitution can occur when these products continue to improve in quality or cost, and even implement some of this product’s functionality in them. This can be resolved by including this product with another product that is popular, and observing other similar products.*

*In terms of threats to cyber security, there is a definite risk as this product is software. This product can be hacked or attacked in some manner where private information can be accessed by unauthorized individuals. Some of this information includes information regarding teachers, students, and TA’s. It could contain names, what school they are associated with, what their association is, and ID numbers. This can make attacking school databases easier for these attackers as they’d receive information that is potentially in these school databases. This risk should be taken seriously. This can be resolved through the use of password systems, having support for encryption, and only offering this product to organizations that have dedicated programmers that can keep this product secure.*

# **Vision of the Solution**

## **Product Idea Statement**

*The program will take in one pdf file with multiple tests in it. This means, for a five page test, every five pages is the fifth/last page of each test. You will input how many pages are in a test, and how many questions there are. The professor will then go through each page and drag a box over each question/answer location for the entire test. Using that box, it goes question by question (all the 1s, then all the 2s, etc) of each test for grading.*

*The grading is not automatic, the professor will be shown the question that is being graded and have an interface where he can type specific feedback for a question and how many points they’re going to lose to be reused on other students answering the same question. There will also be a variety of “canned” answers, but these will be very general. Out of bounds support is needed, so if an answer is trailing off the specified box, they can go to the entire page of that question and then return to the test grading software.*

*After an initial release more features may be added like allowing two or more people to run this software at the same time and thus be able to grade with a TA, as well as a barcode recognition to be able to automatically gather information.*

*Once the tests have been graded, the software will generate a table of scores as a pdf, and a pdf for each student. This pdf for the student will be as many pages as were questions in the test, and list their feedback that they received on each question (1 question per page).*

## **Vision Statement**

*Test grading is a time consuming task that professors have to go through. In a more online age, more and more often tests are being submitted through web based programs leading to digital files being graded. In order to avoid printing all the tests, and to more efficiently grade it is proposed that a piece of software that can help a professor grade a PDF file while also speeding the process up by presenting the question in a more efficient order.*

## **Major Features**

*FE-01 : Import PDF - Users will be input PDF through an open button or drag and drop*

*FE-02 : Set test settings - Users will be able to set how many pages and questions there are in the test.*

*FE-03 : Visual question marking - Users will be able to mark questions by dragging a box around the answers to questions, this lets the filter through later easier.*

*FE-04 : Grading questions - Users will be able to go through the students' tests in order of question instead of test order, making grading faster.*

*FE-05 : Export - Users will be able to export marked tests to a specified folder*

*FE-06 : Overview - Users will be able to export an overview of the test to a specified file.*

## **Assumptions and Dependencies**

*Most dependencies will stem from libraries and support available for working with PDFs in the applications toolset of choice. The features of these libraries will likely decide the features of the application. Some libraries that look promising are PDF.js and pdf-lib for Javascript, or Java PDF Library and iText for Java.*

# **Scope and Limitations**

## **Scope of Initial Release**

*The initial release will include all the necessary components to import tests, set up and grade said tests, and export tests and overview data.*

## **Scope of Subsequent Releases**

*Once the initial release is finished expanding to add things like multi user support and more features will be the next step.*

## **Limitations and Exclusions**

*No limitations or exclusions have been currently identified.*

# **Business Context**

## **Stakeholder Profiles**

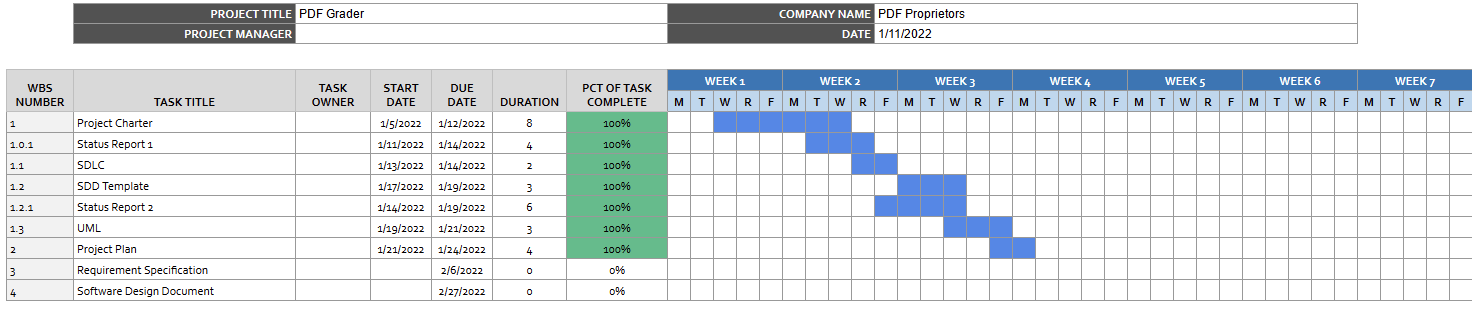
| ***Stakeholder*** | ***Major Value*** | ***Attitudes*** | ***Major Interests*** | ***Constraints*** |
| --- | --- | --- | --- | --- |
| *Professors* | *automation of tedious task, organization* | *desire for streamlined way to grade many structured tests* | *simplicity/ease-of-use, basic features to work reliably, multiplatform* | *software should be lightweight, offline* |
| *Teaching Assistants* | *automation, task clarity* | *excited by fast way to do grading* | *ease-of-use, robust UI* | *free software* |

## **Operating Environment**

*The operating environment of this application will start small and can grow as needed. For the initial release, the application will consist of compiled executables with no internet connectivity or ability to share data. Once the project is ready to expand functionality in order to have multiple people grading or even saving work, this will need to be reevaluated to decide how to save that data and re import it for later use, this could be saved to a server, or in order to be self sufficient, it might be wise to save the data to the PDFs as metadata. For the initial release there is no need for any substantial data protection or security measures because the application will run on the device and data will not leave the device. As the application gets more complicated, security may be something worth revisiting.*

# **Scheduling**

## **Gantt Chart**



## **Key Milestones**

* *File import*
* *File viewing*
* *Marking Questions*
* *Grading Questions*
* *Export*

*More milestones TBD after initial release.*

## **Resource Assignments**

*Resource Assignments TBD*

## **Individual Responsibilities**

*Individual Responsibilities TBD*

# **Deliverables**

## **Software/Hardware**

*Compiled executable for various platforms. E.g. MacOS, Linux, Windows*

## **Documentation**

*Documentation of the code will be provided in order to expand on or fix the underlying systems of the software. This way, if new requirements come up, a new team could potentially pick up the project.*

## **Key Presentations**

* *Final presentation*