Vision Statement

Project Title: Automated 3 Way Match (tentative)
Team Name: $-flow

Members:	**Email**:
● Millan Batra [Lead]	millanbatra@umail.ucsb.edu
● Yoon Lee [Scribe]	yoonlee@ucsb.edu
● Dennis Fong	dfong@ucsb.edu
● Alexander Kang	alexanderkang@ucsb.edu
● Alexander Yuen	atyuen@ucsb.edu
Elementum Mentors:	
● Minh Chau	minh@elementum.com
● Chas Honton	chonton@elementum.com
● Ken Baker	kbaker@elementum.com
● Tony Huynh	thuynh@elementum.com

Problem Description

The complex flow of documents in the purchasing process has progressed to prevent fraud. However, this process continues to have many manual steps which introduce errors and can delay planners, purchasers, and accountants by up to 24 hours.

Our project aims to reduce this 24 hour delay by automating this complex process. This includes mailing and emailing Purchase Orders to the seller, entering Sales Orders into the buyer's' Enterprise Resource Planning (ERP) and matching the sales invoice with purchase orders and shipping invoices to authenticate the payment.

In order to achieve this reduction, we will digitally scan an image, extract its text, make sense of the data, find referenceable invoices, and map the data back to the ERP system.

The current pipeline imposes up to a 24 hour delay before the data is available online and has the potential for human error. This is due to a large amount of manual input when attempting to match a PO, SO, and SI.

We will automate parts of this process, ultimately solving these problems which, as of today have not been addressed. In doing so we will save a great deal of time and money for all parties involved in the pipeline.

Outcome

By the end of this collaboration, we will have completed a mobile app which will help reduce the errors and delays associated with the purchasing process. Our mobile application will allow users to take pictures of invoices and upload this data to the cloud. Once there, we will match the document to related documents and map this back to the customer’s ERP system. This will reduce the amount of errors related with the purchasing process because users will not have manually enter the invoices into their ERP system. It will increase information availability since related invoices are matched to each other and there is a smaller upload delay.

Design

Once we finish a draft of our PRD, we will have a general idea of what our system architecture will be. With that, we will also have drafted versions of our system models and class diagrams. The design will still be subject to change as we start implementing our prototypes and due to completing a draft of the PRD early.

Milestones

By the end of sprint 1 we will have completed a draft of our PRD and have all of our tools ready for use. Familiarize ourselves in React Native, OCR Software, and other tools.

We’ll divide and conquer by starting our backend and frontend and develop a working prototype. Also, we will have begun looking into how the object character recognition(OCR) will be done for our documents.

Platform and Technologies

For our mobile application, we will be using React Native for the mobile view. Tentatively, our backend will consist of Java and PostgreSQL. Machine learning related work will be done in Python using TensorFlow. We will be using tools from the AWS sandbox environment such as Lambda, EC2, and RDS for PostgreSQL. For versioning and issue tracking, we will use Bitbucket linked together with Jira for Sprint planning.