## SDEV220\_teamC\_2023F\_Final\_Project\_Launch

Updated 11/18/2023

Team members:

Mark Atkins [matkins@Ivytech.edu](mailto:matkins@Ivytech.edu) Indianapolis, IN

Joshua Hoover [jhoover82@ivytech.edu](mailto:jhoover82@ivytech.edu) Fishers, IN

Derek Kolb: [dkolb1@ivytech.edu](mailto:dkolb1@ivytech.edu) Medellin, Colombia

### 1. Communication plan:

Discord group: SDEV220\_TeamC: <https://discord.gg/NYFJEY2B>

Zoom meetings: <https://ivytech.zoom.us/j/99454358360>

Saturdays 10 AM EST or 6pm, starting 11/11/2023

We meet weekly via Zoom, and will talk more frequently via Discord and/or email.

All members will “watch” their tasks in the trello board, update trello board to move their tasks through the workflow.

Members will post in GitHub repo the code and code review feedback.

### 2. Project description:

(Fictitious) Client: “ a local company”

Comlux: Derek’s former employer <https://comlux.com/completion/about-us>

* + Other employers considered but not chosen:
    - Abbot labs: Joshua’s employer
    - IvyTech: a fin aid or enrollment student GUI

Scope of the system:

* + Prototype, simple demo of concept
  + Graphical user-friendly interface
  + Uses three or more classes
  + Utilize collections such as lists, tuples, sets, and/or dictionaries
  + Runs with no syntax or runtime errors and produces the correct results
    - Results based on written test plan
  + Documentation: the proposal (this doc), class diagram, and a report Gdoc of results with sample output, including images of GUI

Purpose of the system:

* + Inventory control, tracking (like Trackit @ Comlux @ indy airport)
    - Barcode reader of parts and user badge #s (just emulated, not really implemented), keeping track of aircraft parts received, removed from, returned to storeroom for aircraft repairs. Parts may include new and used (removed from the aircraft)?
    - TENTATIVE classes:
      1. Part
         * Attributes:Part unique ID (S/N?), Manufacturer, Model, Name, Description, Supplier, history (list of dates, conditions, and locations)
         * Methods: Set\_location, Get\_location (maybe unneeded, if location not private), Get\_history
      2. Storage (Stockroom)
         * Attributes: location, employee list, Inventory (list of IDs and date added), etc..
         * Methods: Get\_part\_data, Get\_inventory, Receive\_part, Dispense\_part, other TBD

3. Employee

* + Attributes: Name, employee ID #, role (storeroom worker, storeroom customer, shipping/receiving, etc), security clearance, etc..
  + Methods: Get\_employee\_data, Set\_clearance, others TBD

4. GUI (probably a super class, using Tkinter package)

* + - * + Dialog box for stockroom person
        + Dialog box for customer
        + Attributes: TBD
        + Methods: TBD
    - UML class diagram (rough, preliminary):



* + We brainstormed and considered the following also, but did not choose these. The decision may be revisited if necessary:
  + Budget tracker
    - a personal finance budget tracker 50/30/20 (per Derek)
  + HR data management system
    - Name, id, dates of hire, position, job history, pay, supervisor, subordinates, etc
  + Multi-File editor (for zybooks.com)
    - Simultaneous editing of selected files in single local folder ***or Gdrive folder*** (via Gdrive API)
    - Find string, replace, count # of occurrences
    - Somewhat like Bulk Docs Find and replace Chrome add-on [Bulk Docs Find & Replace - Google Workspace Marketplace](https://workspace.google.com/marketplace/app/bulk_docs_find_replace/57575149967)
    - Does not look feasible to edit Gdrive files in Python. Pydrive2 package allows upload, download, creation of files but not calling the editor
      1. Could download the selected files (as .odt or .rtf ??), edit locally with Python, upload again, but that is messy. The ,odt translations might mess up the file.

Complex code, dealing with .odt formatting also.

* + - 1. Google scripts allow customizing the Gdoc editor with menus or sidebars, but the script must be in Javascript, can’t be Python
  + Graphics file creation app like MSpaint or Scratch or…
  + Image processing tool, minor subset of Adobe photoshop, to recolor, smooth, sharpen photos

### 3. GitHub repo:

[*https://github.com/mdatkins/SDEV220\_project\_teamC*](https://github.com/mdatkins/SDEV220_project_teamC)

We plan to use the collaborator features of GitHub.

### 4. Trello board: SDEV220\_project\_teamC\_2023F

<https://trello.com/b/jh9HHOrf/sdev220projectteamc2023f>

Team member roles/responsibilities:

~~Mark Atkins: project manager + ⅓ of code + ⅓ of testing + ⅓ of final report~~

Joshua Hoover: GitHub repo, ⅓ of code + ⅓ of testing + ⅓ of final report

Derek Kolb: GUI, ⅓ of code + ⅓ of testing + ⅓ of final report

We have not yet architected the code, so we will further clarify which “⅓” each person does, later when the code is better specified.

### 5. Code modules (objects, functions)

Main

Starts GUI, runs User test, runs stockroom test

Each class

User (technician)

Stockroom