**WORK PLAN**

1. Understand the Project’s Objectives and its Data (Frame the Problem)
   1. Provide an Introduction describing:
      1. The problem the company is trying to solve and why they are trying to solve it
      2. Any other information related to the “story” of the project
      3. The definitive objectives of the project (criteria for success)
         1. AUC-ROC >= 0.85
   2. Provide a Section of all imported modules
2. **What data is included in each dataset, and how do they relate to each other?**
   1. Descriptions of each column in each set
      1. Name,
      2. Data Type,
      3. % of missing values,
      4. Noisiness,
      5. Useful for the task?
      6. Type of distribution?
   2. Observe: sample rows, shape, duplicates, nulls, dtypes, describe (all)
   3. Document any discovered issues or concerns for pre-processing.
   4. Do any Implicit erroneous values exist in each column – especially the categorical vars?
   5. Can any columns be decomposed into more features, or new features created to improve existing information?
   6. Form Conclusion/Summary for pre-processing – based on step 2.c
3. **Pre-Processing**
   1. Merge the datasets on **CustomerID**
   2. Execute fixes from Step 2.c
   3. Feature Engineering: Use notes from exploration to reduce or create new features.
4. **EDA**
   1. Distributions of Numerical Features (Histograms, boxplots)
      1. Individual distributions (*If not done already in exploration, likely not*)
      2. Churned vs Retained
   2. Distributions of Categorical Features (Bar charts)
      1. Churned vs. Retained – *Using filled charts*
   3. Distribution of Target Class
      1. Is there an imbalance?
   4. **Questions to ask:**
      1. Are the monthly payments different among Target classes?
      2. What is the customer share per service type?
      3. When were the customer start dates and end dates – any patterns there?
      4. Any specific features we want to compare the target class in?
      5. Observe correlation matrix of numerical features and target
   5. Write a conclusion of all findings.
5. **Modeling**
   1. Reuse functions and processes from previous projects – attempt to improve
   2. Function for preparing and training models?
   3. Function for evaluating
   4. Function for loading and saving models – saves time
   5. Drop Any determined unnecessary features lingering from EDA
   6. Split data into features and target
   7. Perform OHE and feature scaling – use standard scaler
   8. Apply decided imbalance correction
   9. Split data into train, validation, and test: (80:10:10)
   10. Train and evaluate the models using functions if applicable
6. **Final Model Selection**
   1. Choose best performing model
   2. Evaluate Final Model
7. **Conclusion**
   1. Provide overall conclusion

**QUESTIONS FOR LEAD AND SELF:**

* Is there an imbalance in the target class and how would we like to address it?
  + Upsampling, downsampling, bootstrapping, weights
* How will our solution be used?
* What are the current solutions, if any?
  + Are there anay recommendations on specific models that have been successful in similar prediction tasks?
  + Can I reuse any experience or tools? **Yes I have worked on a churn problem before. In addition, I can reuse many functions and processes from previous projects to aid me in code development.**
* Is the performance measure aligned with the business objective?
* Are there specific features that require in-depth exploratory analysis? Which should we prioritize?
* How would you solve this manually? **Look for patterns.**
  + Look at customer balances, tenure, customer spending/debt on average;
  + what about other factors that drive customers to leave? **Customer support, User friendliness of app, our prices/packages/fees compared to competitors**
  + Any patterns in the data that show patterns of customers staying? Factors that encourage customers to stay: ------
* Any preferences on how to handle missing values?

**MODELS TO USE & CRITERIA**

* Constant Dummy, Tree-based, logistic reg (linear would perform poorly), a variety of bossting methods: ada, cat, xgb, lightgbm, histgrad, ***Neural Network?, stacking ensemble??***
  + - ***For the stack – blend boosting and log-reg?***
* AUC-ROC and Accuracy will be metrics
  + **AUC-ROC OF AT LEAST 0.85, 0.88 IS THE GOAL**
* **Boosting models will be our best friends here** (except maybe the NN) but **overfitting** is a concern so we will need to prioritize **regularization** more so than previous projects