Week_1_df
You were recently hired to be a part of an IT consulting company that provides IT solutions to big businesses. Your team is presented with different projects and your supervisor asks you to apply SDLC (requirements, design, etc.) to your project.

- What would be your recommendation based on the SDLC methodology? Justify your recommendation.
- Try to sell your idea to your boss and your team of experienced software engineers. Describe the pros and cons of your recommended approach. Give a realistic estimate of the time, cost, and feasibility of your proposed recommendation.
- Would you recommend Git or GitHub for a management tool for the design process? Why or why not?

Here is the project:

A bank wants to provide higher security and ease of access to its customers. The bank wants customers to be able to use their fingerprints to access ATM services. What is the best approach to creating this fingerprint-based ATM system? Apply the SDLC methodology to this scenario.

Include in your response:

- Business requirements – what is the problem? What do you need to create?

Currently a regular ATM system requires a card. User types a PIN code and can do a number of operations. A bank continuously works on customer’s satisfaction and security. One of its innovations is a fingerprint based ATM machine, which allows use of a person's fingerprints. Our solution should contain two parts: a hardware part, fingerprint scanners, ATM machine and a software. Software include: Operating system, drivers, GUI (app), monitoring, alarming and traceability.

- The proposed solution – code something from scratch or seek out a vendor or a combination of both? Be specific and use actual programming languages, programs, or software vendors.

Our solution will be based on AI/ML, computer vision. We need data, Kaggle can give it to us. This data is a lot of fingerprints to train our model. So, once a fingerprint is scanned, our application compares with the database and will allow or reject entering to the user’s account. We are going to use open source libraries, solutions and frameworks, like tensorFlow. Main language will be: Python and as a backend Java/Go. Application written on React. Database is Postgres SQL.

- SDLC method – what is your approach? Why? What are the pros and cons of your approach?

It is nice to have followed the waterfall methodology however the bank wants to see our progress as much as possible. In addition, changes will happen from time to time. Agile + Scrum should make delivery our solution more gradually, more rapidly and more organized. Our Spring will be each two weeks, with planning on this period only. After each Spring, bank representatives will be on retrospective to get feedback. I agree that if our team was using another SDLC it takes time to accumulate the knowledge and style. At some point lack of documentation can make life though. This methodology also requires more professional stuff.

    - Development – how is development done? Is this feasible? What is the timeline for deployment? What is the estimated total cost of development and deployment?

Development done by chunks, each two weeks, plus TDD, i.e test driven development. Each class, business logic requires to be covered by unit tests. Each Sprint we deliver one piece of working software, it can be a UI design, UI form or CSS. Our development has two environments: QA and PROD. Each comment will trigger CI/CD with integration tests, A/B tests. After a commit is checked and merged, our PROD env will be redeployed. Our customers have access to the QA and PROD env. We also will give access to better testers.

Red Teams also will be part of the cycle.

What we are going to deliver: GUI (1 month), AI/ML (model training - 1 month), Backend (1 month) + 2 month as stabilization, bug fixes or change adoption. Depending on available resources: Security experts, GUI experts, Data scientists + Backend developers say each team works independently. So, we can be in production in three months. If each team has three team members, with a monthly salary - 10K + healthcare + insurance requires 250K for us. In addition, office rent, computers, IDEs and other licenses may cost - 16 * 2K + 10k + 30k = 72K.

Plus snakes, unpredicted costs totally we need about 350K.

- Would you recommend Git or GitHub for a management tool for the design process? Why or why not?

I would recommend using GitHub as a source control and collaborative tool. For Sprint planning and management I would recommend Jira. For UI - figma, for any kind of diagram would take draw.io. On the other hand, atlassian claims that software development can be seen as branches in git. In any case I would work with github + Jenkins + Argo/Trevis. Jenkins has a webhook and listens to the github events. Based on the type of event it will trigger a job. Argo will run this job on pods and deploy it to specific environments. Jira also has a link to git commit to keep track of history/progress and status. In addition each github repo contains README and other md files which could be transformed to the documentation.

    - References – at this point, as a graduate, you are the expert on your team so you need to provide sources that support and back up your claims.

References:

1. Weider Biometric ATMs. https://atmdepot.com/biometric-atms/
2. 14 pros and cons of Agile Methodology. (March 11, 2023) https://www.indeed.com/career-advice/career-development/pros-cons-of-agile-methdology
3. Tensorflow - https://www.tensorflow.org/
4. Fingerprint datasets: https://github.com/robertvazan/fingerprint-datasets
5. Kaggle - https://www.kaggle.com/code/kairess/fingerprint-recognition
6. React - https://dev.fingerprint.com/docs/fingerprintjs-pro-react
7. Fingerprint app git repo: https://github.com/fingerprintjs/fingerprintjs-pro-react
8. Arizona department of public safety: https://www.azdps.gov/services/public/fingerprint
9. Fingerprint rolling certification - https://www.azdps.gov/services/public/fingerprint
10. Git makes software development easier - https://www.atlassian.com/agile/software-development/git
11.