Contributing to an Open source project

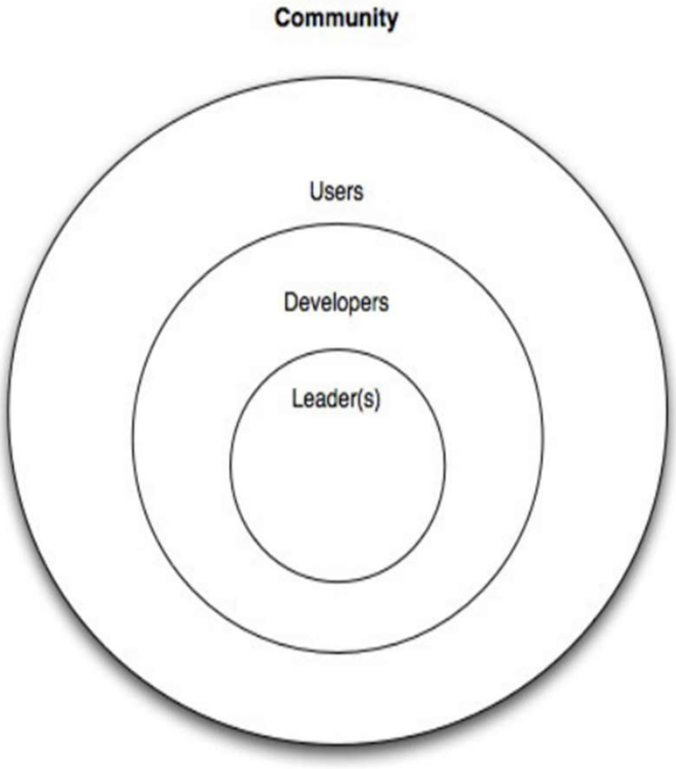
Sir , i gave 20 rupees for printOut

Contributing to OSS

- To an existing OSS Project

- To start a new OSS Project

OSS Community: many roles to contribute



OSS is community driven

- Many roles to play/involved with
  - Depending on your interest
    - Can take more than one role[happens often].
- Roles to play:
  - Project Lead/Main Developer [similar to CEO]
    - When you start your own project
    - Open source an existing project seeking contributions
  - Programmer/developer/contributor
    - Implementing:
      - Features
      - Fixing bugs

Other Contributor roles

- Designer
  - System designer
  - UI/UX designer
  - Web design
  - Art work design
- Documentation Writer
  - Project documentation
  - Help content
  - API documentation
  - Installation guides
- Translator
  - Translating project manuals etc to native languages.
- Active User/Tester
  - Using the software
  - Giving feedback
  - Reporting bugs

Some other ways of contributing[other than coding]

- Writing tutorial on how to use the product
- Creating news letters
- Planning the events[workshops, meetups, conferences]
- Refining existing designs[navigation, style guides for visual layout]
- Contributing to forums
  - Answering questions
  - Moderating discussion boards etc
- Contributing in organizing the project
  - Restructuring the project layout
  - Linking duplicate issues
  - Labelling etc.

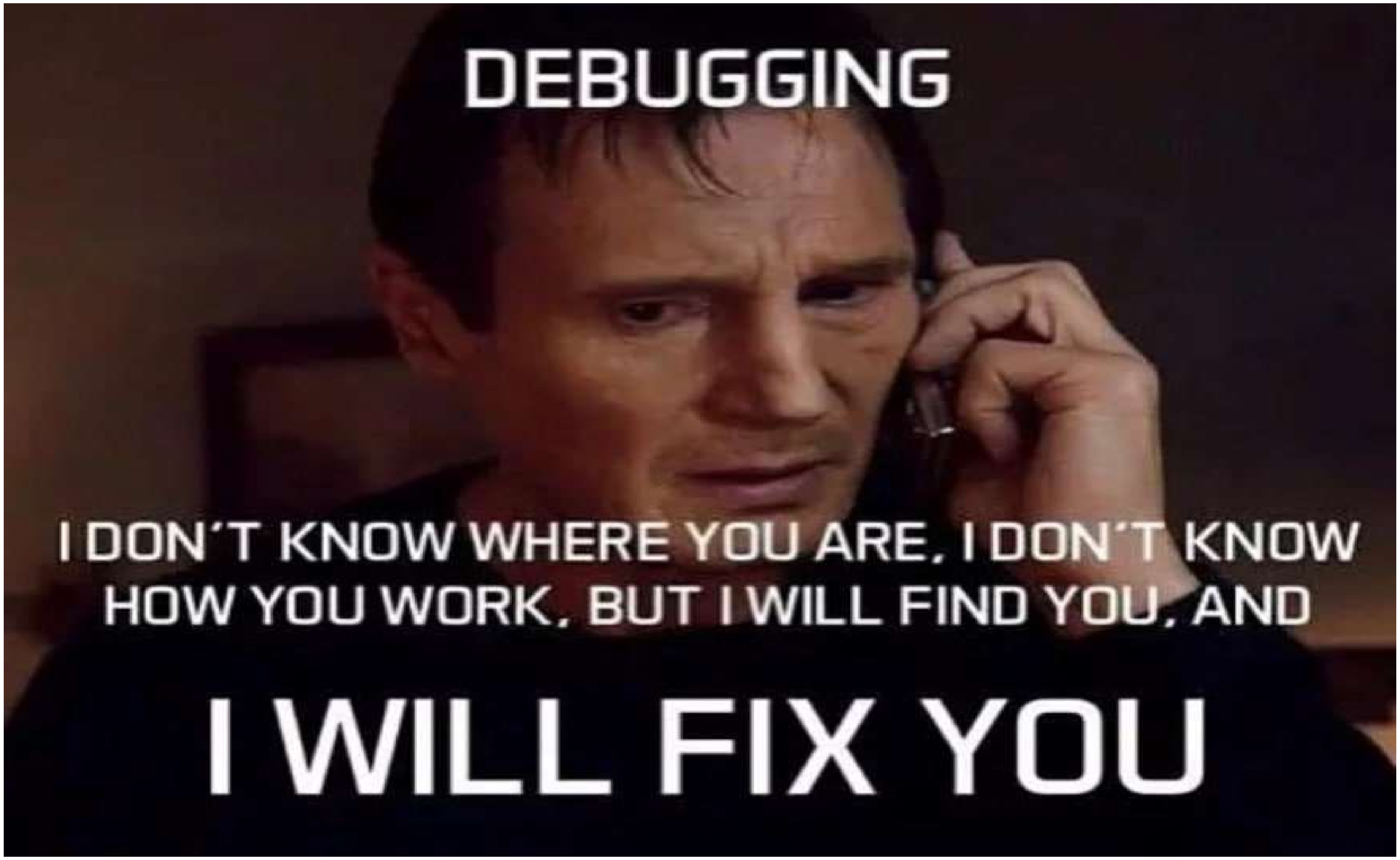
I am Ready to contribute to OSS Project

Familiarizing with OSS project

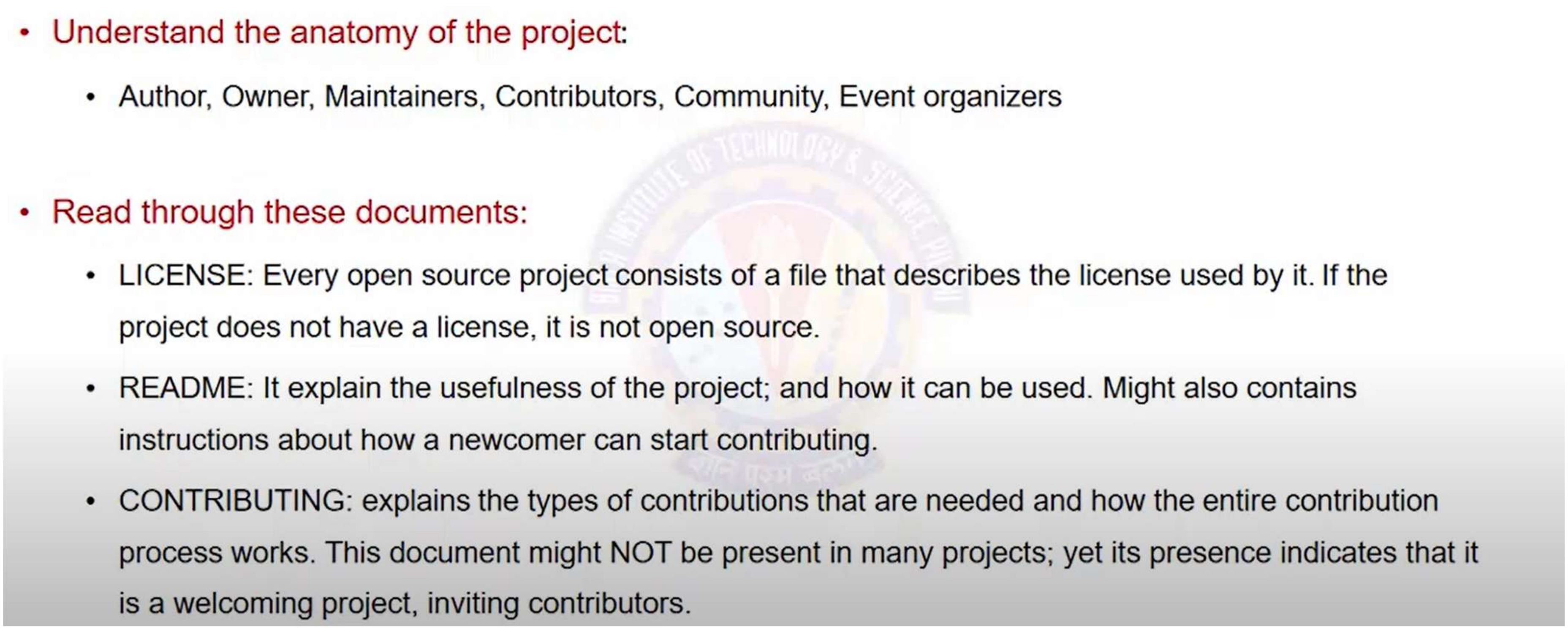
- Prerequisites:
  - Learn a programming language
    - Start with any language you know.
  - Get used to a VCS[version control system]
    - Git, SVN, Mercurial, CVS.
      - GitHub is the most popular.
  - Search for an active OSS project in your chosen language.
    - Not much benefit to contributing to a dead OSS project.
    - Navigate and explore the repository.
    - Read the documentation[“readme” etc]
    - Go through the license information
    - Is there any CLA to be signed?
      - Go through the CLA document in detail.
      - Assess whether you can sign the CLA or not?
    - Join the communication channel of the project[IRC/discord/slack]

- Start with beginner friendly issues.
    - Check the labels
    - Look at the pull requests
- Understand the structure of the project
    - Author/owner, contributors, community, maintainers etc

Getting/familirizing into an existing OSS project


DEBUGGING
I DON'T KNOW WHERE YOU ARE, I DON'T KNOW
HOW YOU WORK, BUT I WILL FIND YOU, AND
I WILL FIX YOU

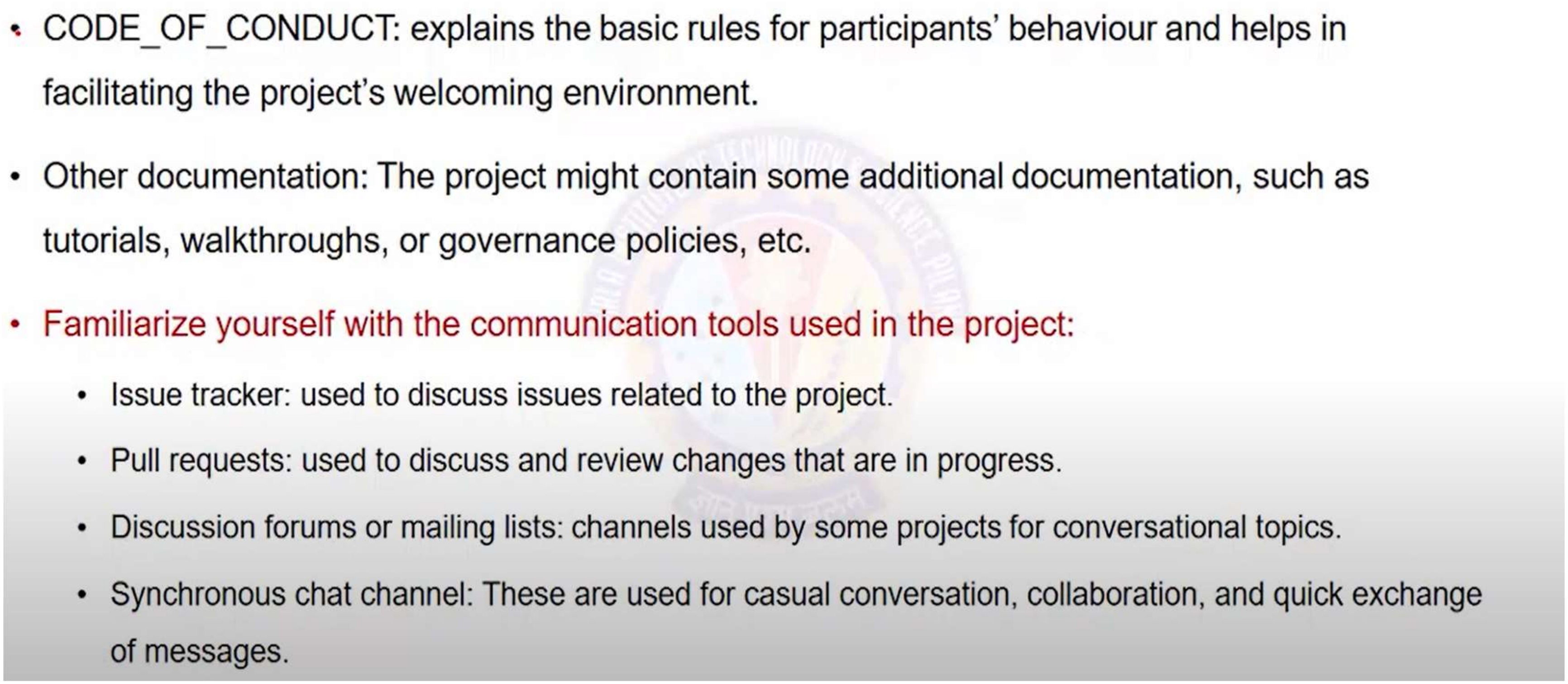
Familiarizing yourself with an existing open source project



- Understand the anatomy of the project:
  - Author, Owner, Maintainers, Contributors, Community, Event organizers

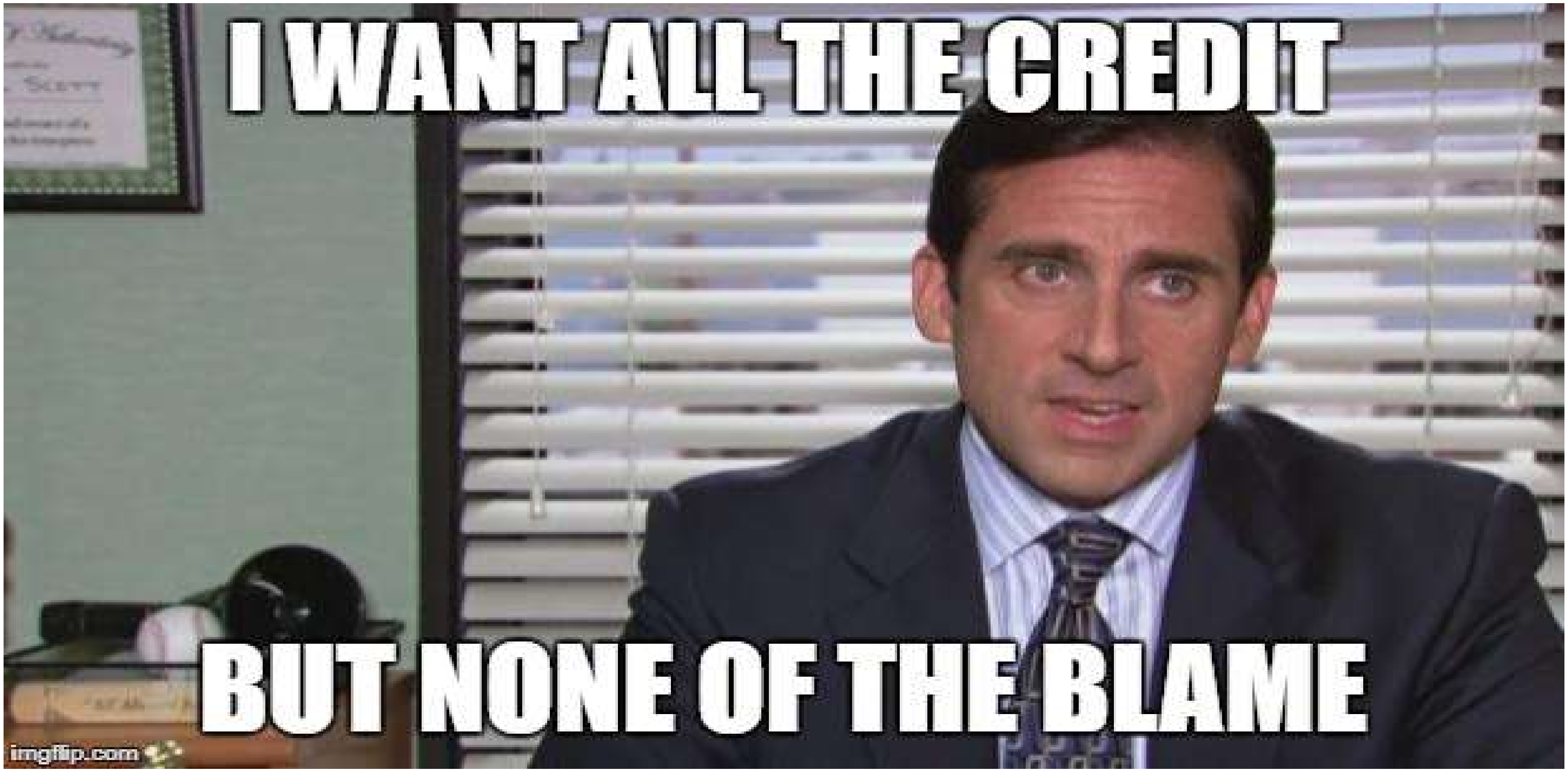
- Read through these documents:
  - LICENSE: Every open source project consists of a file that describes the license used by it. If the project does not have a license, it is not open source.
  - README: It explain the usefulness of the project; and how it can be used. Might also contains instructions about how a newcomer can start contributing.
  - CONTRIBUTING: explains the types of contributions that are needed and how the entire contribution process works. This document might NOT be present in many projects; yet its presence indicates that it is a welcoming project, inviting contributors.

Continued…

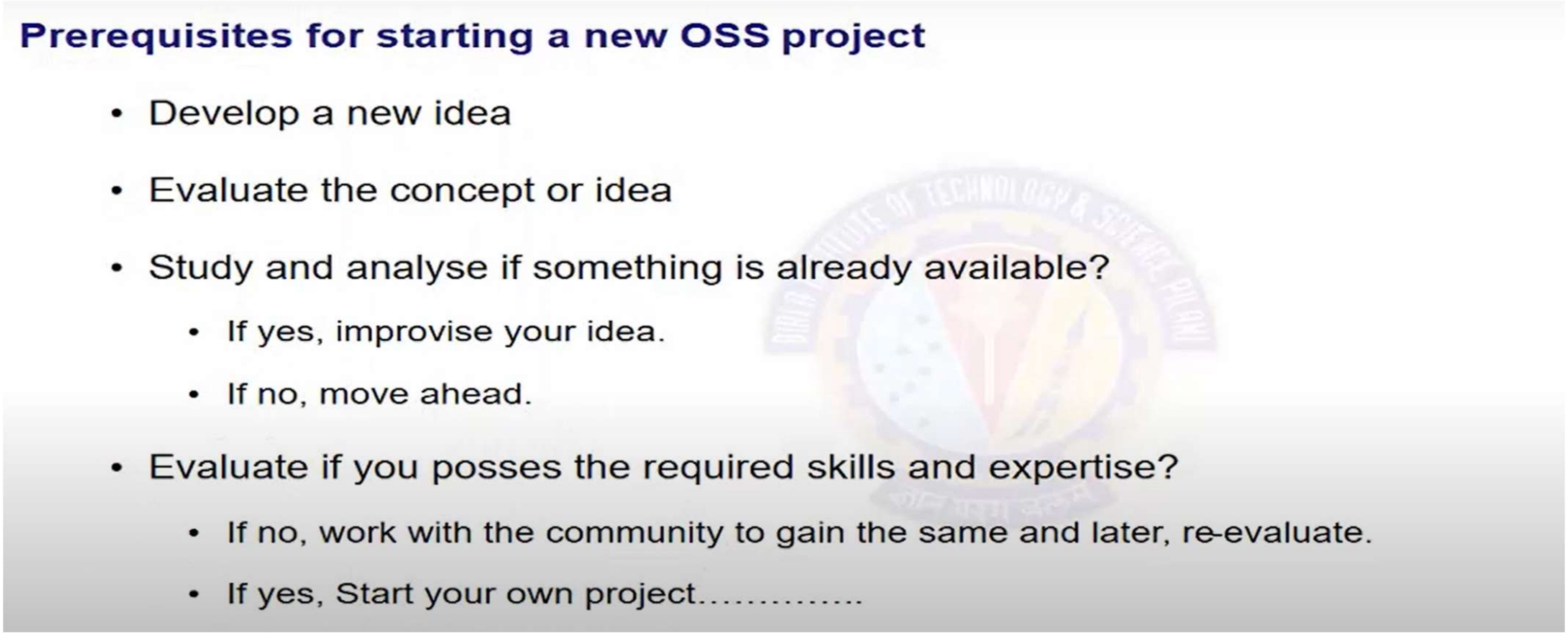


- CODE_OF_CONDUCT: explains the basic rules for participants’ behaviour and helps in facilitating the project’s welcoming environment.
- Other documentation: The project might contain some additional documentation, such as tutorials, walkthroughs, or governance policies, etc.
- Familiarize yourself with the communication tools used in the project:
  - Issue tracker: used to discuss issues related to the project.
  - Pull requests: used to discuss and review changes that are in progress.
  - Discussion forums or mailing lists: channels used by some projects for conversational topics.
  - Synchronous chat channel: These are used for casual conversation, collaboration, and quick exchange of messages.

Starting your own…



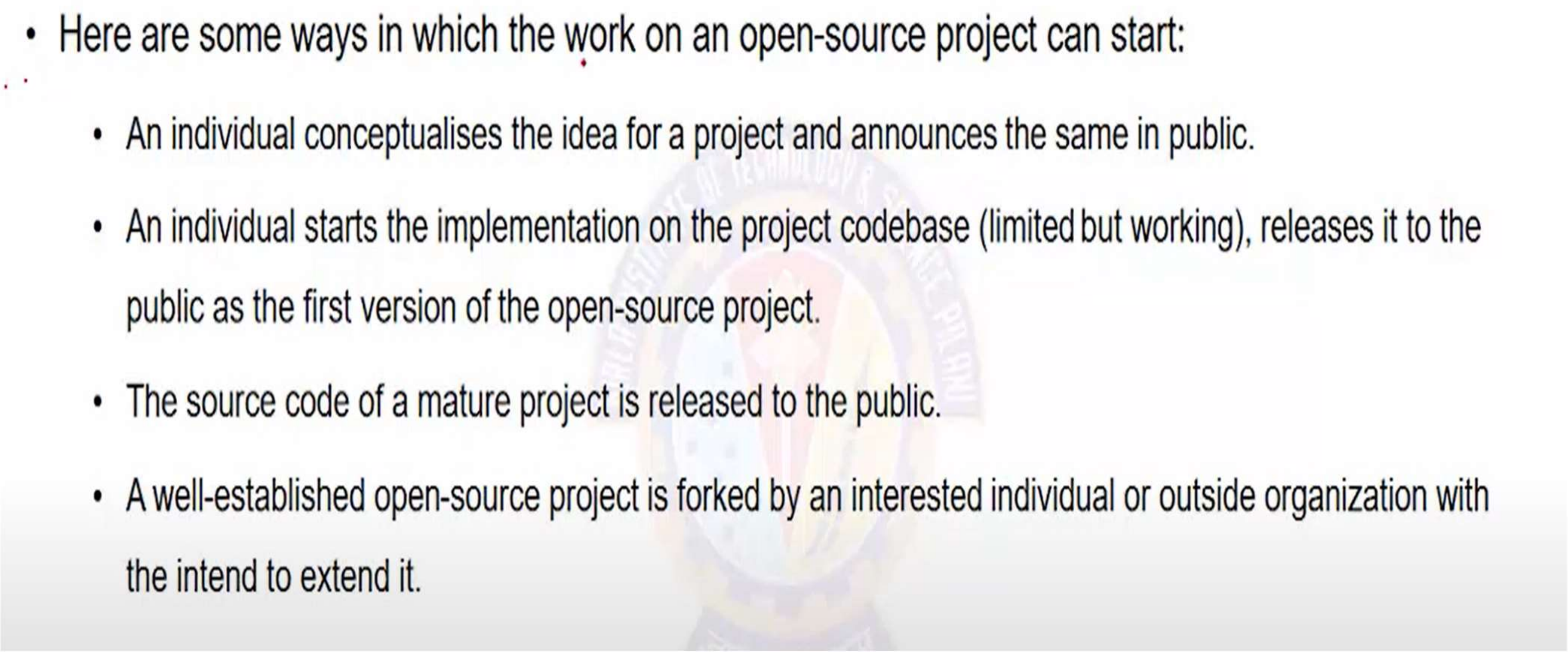
Starting your own Open Source Project



Prerequisites for starting a new OSS project

- Develop a new idea
- Evaluate the concept or idea
- Study and analyse if something is already available?
  - If yes, improvise your idea.
  - If no, move ahead.
- Evaluate if you posses the required skills and expertise?
  - If no, work with the community to gain the same and later, re-evaluate.
  - If yes, Start your own project…………..

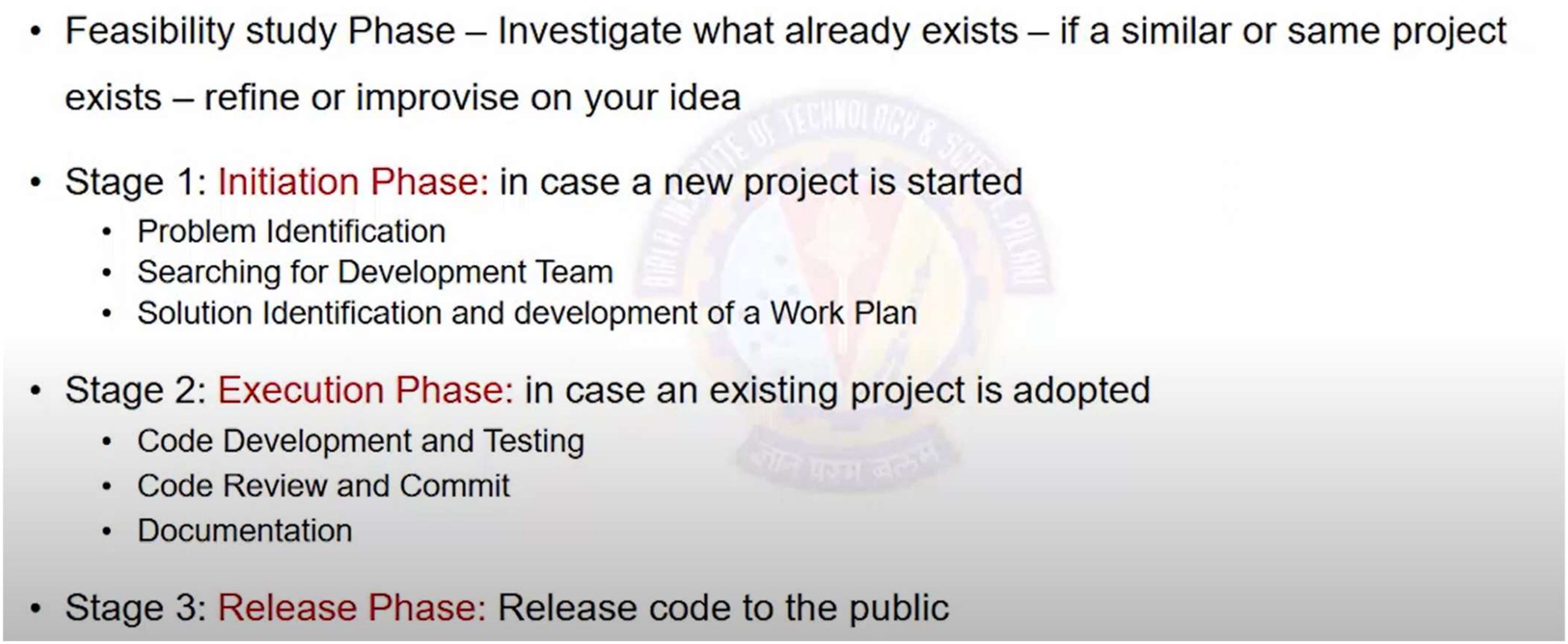
Continued…



- Here are some ways in which the work on an open-source project can start:
  - An individual conceptualises the idea for a project and announces the same in public.
  - An individual starts the implementation on the project codebase (limited but working), releases it to the public as the first version of the open-source project.
  - The source code of a mature project is released to the public.
  - A well-established open-source project is forked by an interested individual or outside organization with the intend to extend it.

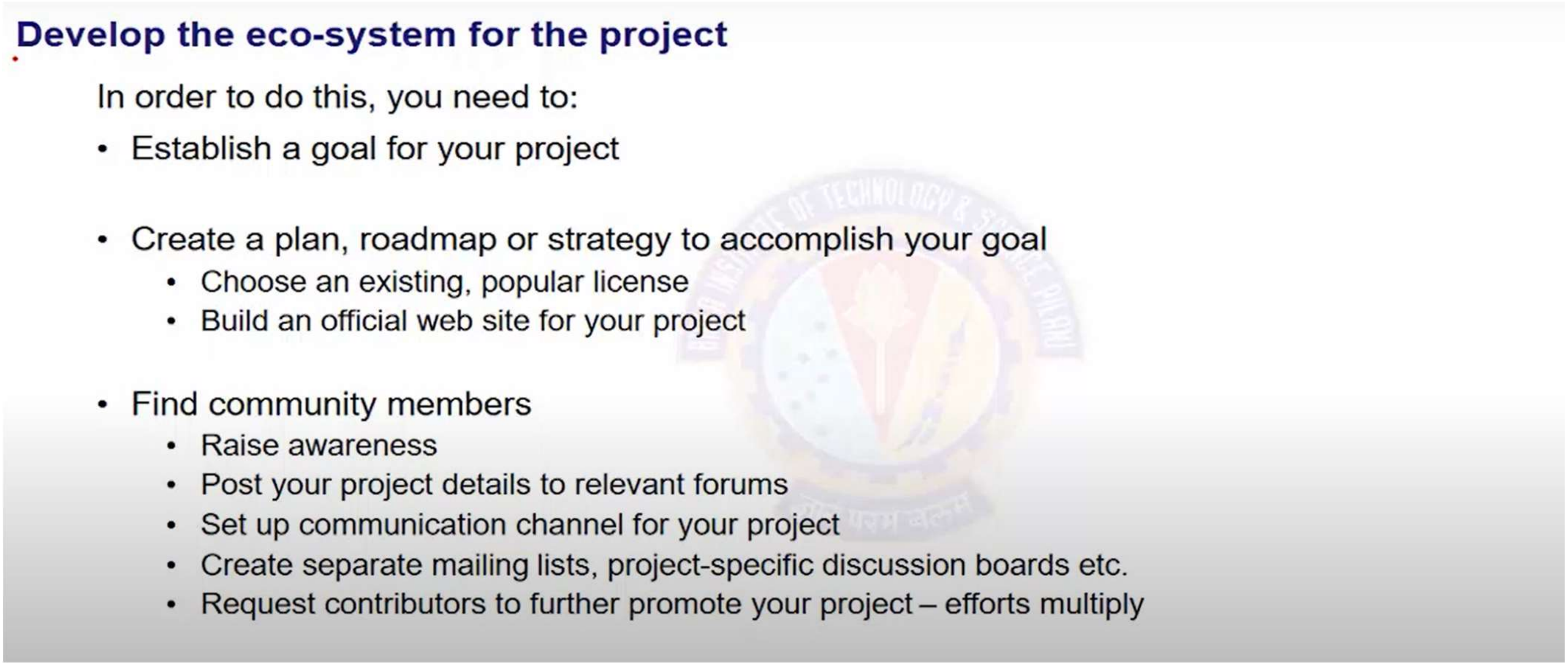
- Best way is to release a partial working project/prototype to the public and then gathering the support.

Process model for developing OSS Projects



- Feasibility study Phase – Investigate what already exists – if a similar or same project exists – refine or improvise on your idea
- Stage 1: Initiation Phase: in case a new project is started
  - Problem Identification
  - Searching for Development Team
  - Solution Identification and development of a Work Plan
- Stage 2: Execution Phase: in case an existing project is adopted
  - Code Development and Testing
  - Code Review and Commit
  - Documentation
- Stage 3: Release Phase: Release code to the public

Other Steps to be considered…



Develop the eco-system for the project

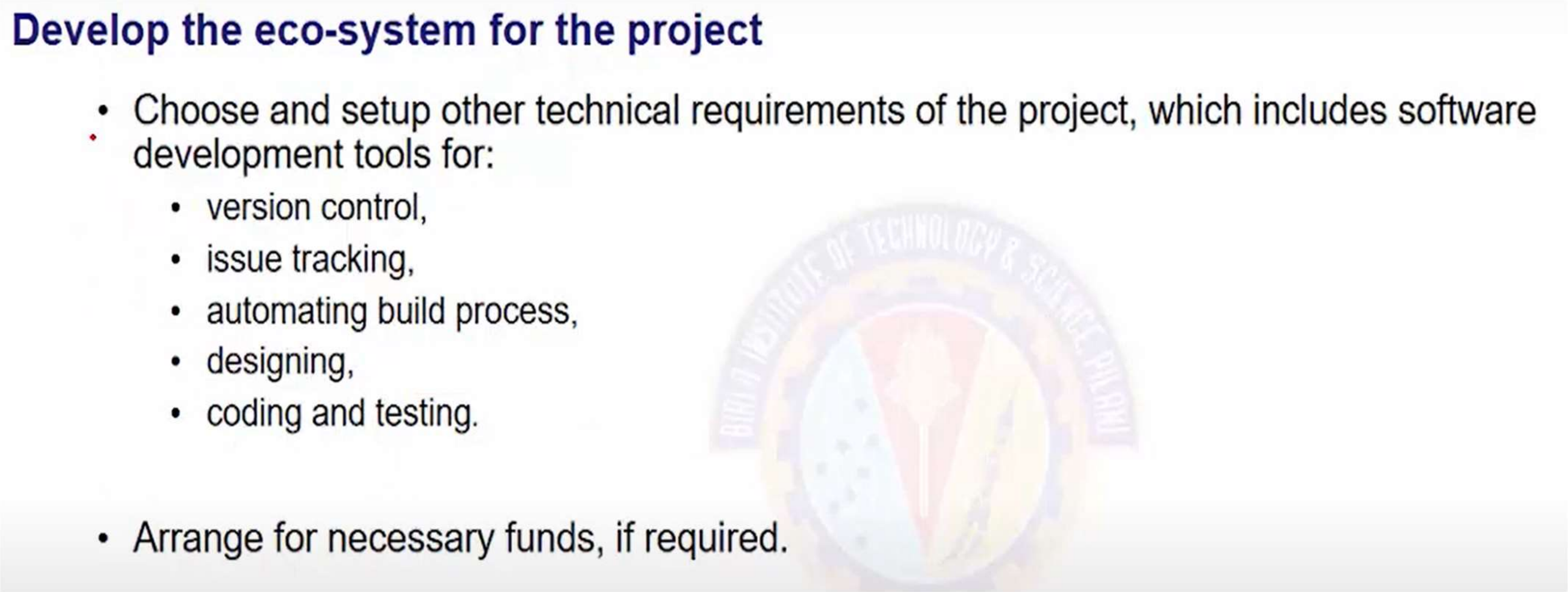
In order to do this, you need to:

- Establish a goal for your project

- Create a plan, roadmap or strategy to accomplish your goal
  - Choose an existing, popular license
  - Build an official web site for your project

- Find community members
  - Raise awareness
  - Post your project details to relevant forums
  - Set up communication channel for your project
  - Create separate mailing lists, project-specific discussion boards etc.
  - Request contributors to further promote your project – efforts multiply

Continued…



Develop the eco-system for the project

- Choose and setup other technical requirements of the project, which includes software development tools for:
  - version control,
  - issue tracking,
  - automating build process,
  - designing,
  - coding and testing.
- Arrange for necessary funds, if required.