QA

QA stands for Quality Assurance. Their task is to test all the functionalities of a product, and to find all bugs and issues that users might encounter while using the product, by meticulously testing it themselves. While doing so, they should document all their testing so that once they find a bug or an issue, they know exactly how it happens, and can report to the rest of the team, that will then use this information to fix it.

They give the green light on whether the product is ready to be released. Due to their extensive use of the product, they can also give feedback on how to improve it.

Their tasks include:

- Writing extensive test plans: what there is to check, and how it will be checked.
- Finding bugs and issues while the product is being developed, by running tests and checking the functionalities. They can and should create tools for automating the testing.
- Keeping track on each bug and issue encountered, and how it was fixed, by using a database of bugs.
- Checking that needed modifications are done according to the specifications, and that issues have been fixed properly.

To do their job successfully, the QA need to be able to put themselves in the users' shoes and think of every way they might interact with the product. It requires understanding how different users might think, from whose who have no idea how to use the product, to those who are already familiar with it and might have specific habits. They need to be thorough, and they must be familiar with all technical characteristics of the product.

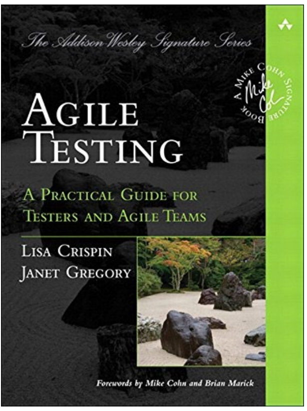
Their main deliverable are: the test plan and the database of bug reports.

https://www.qatouch.com/blog/roles-and-responsibilities-of-qa-in-software-development/

https://www.joelonsoftware.com/2000/11/08/painless-bug-tracking/ (specifically on what is a database of bugs and how useful it is)

Agile Testing: A Practical Guide for Testers and Agile Teams (Addison-Wesley Signature), by Lisa CRISPIN

Fifty Quick Ideas To Improve Your Tests, by Gojko ADZIC


The Addison-Wesley Signature Series
A Mike Cohn Signature Book
AGILE TESTING
A PRACTICAL GUIDE FOR TESTERS AND AGILE TEAMS
LISA CRISPIN
JANET GREGORY
Forewords by Mike Cohn and Brian Marick

Software engineer

They are the specialist in all things technical, and need to make sure that the software is up to date and performing as expected. They are responsible for writing the code, making sure of its quality, and providing directions to fellow team members.

Their tasks include:

- Writing the code and developing the product, including identifying and fixing any technical issue. They can and should use methods such a unit testing to make sure that the code works correctly.
- Participating in the technical design of the product and determining project requirements, which will then lead to suggesting the software architecture.
- Ensuring that the product's quality matches the client's expectations.
- Writing the code's documentation, making sure it follows the convention.

Hard skills are important for this role, as they are the main person in charge of writing the code. They also need soft skills such as effective communication, problem solving, and strong time management.

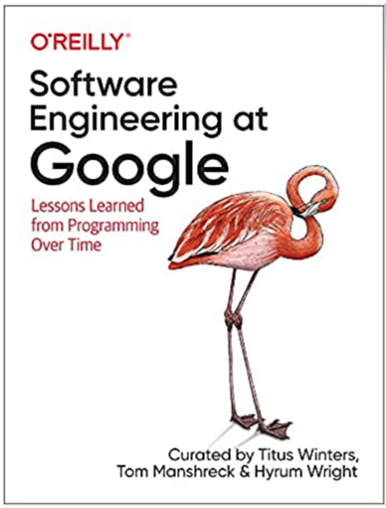
Their main deliverable is: the code and unit tests.

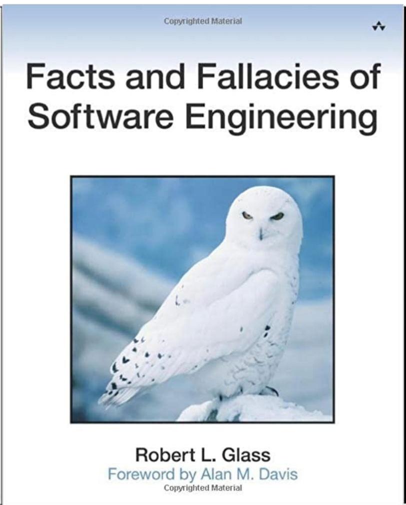
https://www.coursera.org/articles/software-engineer

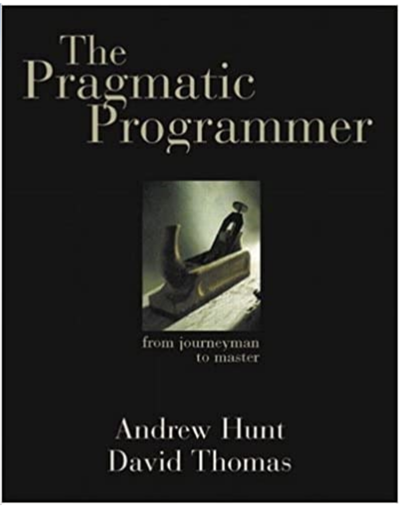
The Pragmatic Programmer, by Andrew HUNT and David THOMAS

Facts and Fallacies of Software Engineering, by Robert L. GLASS

Software Engineering at Google, by Titus WINTERS, Tom ANSHRECK, and Hyrum WRIGHT


O'REILLY
Software Engineering at Google
Lessons Learned from Programming Over Time
Curated by Titus Winters, Tom Manshreck & Hyrum Wright


Copyrighted Material
Facts and Fallacies of Software Engineering
Robert L. Glass
Foreword by Alan M. Davis
Copyrighted Material



Program manager

The program manager is the one making sure that the product will meet expectations upon delivery. While the name is similar to “project manager”, these are two completely different roles. Program managers generally supervise groups of projects, but during ALGOSUP’s project time they will learn how to effectively supervise one project first. They are also in charge of the design of the product: part of their role is to make sure the product/interface is user-friendly, and to reduce the users’ effort while using the product. They need to make sure it is straightforward and intuitive enough to use.

Their tasks include:

- Writing the functional specifications according to the client’s needs and expectations, and possibly re-writing them as the product evolves and the client highlights new needs.
- Handling communication with the stakeholders/clients and the development team: they must create a glossary to make sure everyone each party understands the other and uses the correct terms.
- Leading focus groups where they collect users’ feedback and opinions, and analysing the users’ feedback to highlight their needs, as well as doing a competitor analysis.
- Sharing their findings with the development team so changes can be made; they are the main link between the prospective users (persona) and the team, and need to be able to explain the users’ point of view and needs to the developers.

They need to have a good understanding of the objectives of the program and the clients’ desires and be able to put them into functional specifications and effectively communicate them with the rest of the team. The program manager deals with the outcome of the program, and how any change might affect the result. They are responsible for the integrity and coherence of the program. They need to have good organisational and communicational skills, as well as problem solving skills.

Their main deliverable is: the functional specifications.

https://www.wrike.com/blog/program-manager-vs-project-manager/

Why functional specifications are important https://www.joelonsoftware.com/2000/10/02/painless-functional-specifications-part-1-why-bother/

What makes good specs https://www.joelonsoftware.com/2000/10/03/painless-functional-specifications-part-2-whats-a-spec/

https://careerfoundry.com/en/blog/ux-design/what-does-a-ux-designer-actually-do/

Writing for Computer science (second edition), by Justin ZOBEL

Specification by Example, by Gojko ADZIC

Project manager

The project manager plans, organises and directs the completion of an individual project. While the name is similar to "program manager", they are two completely different roles. They report to the stakeholders on progress and changes made to the plan. They mostly focus on execution and manage functional elements of the project such as meeting deadlines, staying within budget, delegating tasks, and completing deliverables. They identify all the tasks necessary for releasing the product, and evaluate how much work they will represent, then divide the workload based on available resources and establish the critical path.

Their tasks include:

- Managing time by communicating with the team (through meetings, daily check-ins, emails or calls). They need to know the status of every step taken by the team, and to make sure that everyone stays on track.
- Anticipating potential roadblocks and issues, and proactively resolving them or escalating the issue to management.
- Budgeting and allocating resources. Depending on the team's needs, they might be advocating for more or different resources.
- Keep the team motivated and happy, so they can be as productive as possible. They might plan team-building exercises or events.

Their main skills are planning and communication. They need to pay attention to every detail of every other team member's work, and to be flexible enough to be able to reorganise things as needed to meet the deadlines.

Their main deliverable is: the final product.

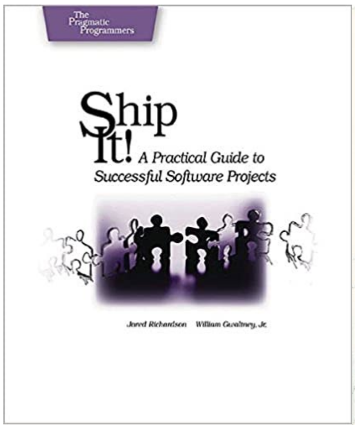
https://www.northeastern.edu/graduate/blog/project-manager-responsibilities/

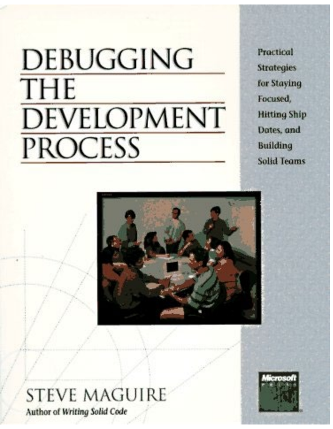
https://www.wrike.com/blog/program-manager-vs-project-manager/

https://www.northeastern.edu/graduate/blog/essential-project-management-skills/

Debugging the Development Process, by Steve MAGUIRE

Ship It! A Practical Guide to Successful Software Projects, by Jared RICHARDSON and William GWALTNEY Jr.


The Pragmatic Programmers
Ship It! A Practical Guide to Successful Software Projects
Jared Richardson William Gwaltney, Jr.


DEBUGGING THE DEVELOPMENT PROCESS
Practical Strategies for Staying Focused, Hitting Ship Dates, and Building Solid Teams
STEVE MAGUIRE
Author of Writing Solid Code
Microsoft Press

Tech lead

Tech lead stands for technical leader. They create the concept for the software and help turning this concept into a plan. They are very involved in the decision-making process of the project, as they have a say in what tools, language, coding standards and platforms will be used. They translate the client's ideas into technical tasks to distribute to the team, according to current technical possibilities.

Their tasks include:

- Interacting with the program manager to understand the functional specifications and translate them into technical specifications, deciding which tools will be the best fitted to develop the product.
- Performing code reviews on a regular basis, to ensure the quality of the product. They work on the prototype development, code and/or technological assessment.
- Writing the architecture diagram that will contain all the information necessary to the software's creation, such as internal and external component and how said components interact and communicate with one another.

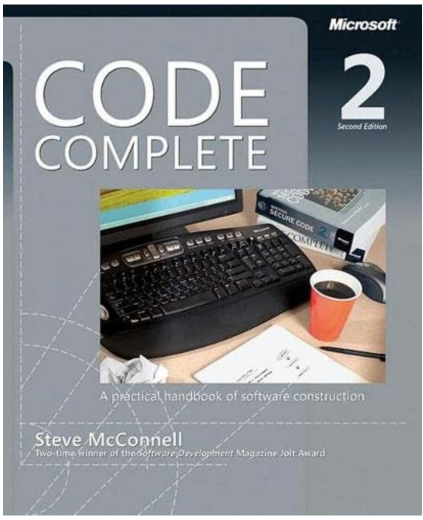
A tech lead should code so they see the development of the product from the "inside". In addition to hard skills, soft skills are very important for this position, as a good tech lead should be an excellent communicator: they need to present their team's work, and to be a link between the development team and others.

Their main deliverables are: technical specifications and architecture diagram.

https://dev.to/thawkin3/lessons-from-a-tech-lead-roles-responsibilities-and-words-of-advice-ldj

Code Complete (second edition), by Steve McCONNELL (old but still relevant)

Software Mistakes and Tradeoffs, by Tomasz LELEK and Jon SKEET


Microsoft
CODE
COMPLETE
2
Second Edition
A practical handbook of software construction
Steve McConnell
Two-time winner of the Software Development Magazine Jolt Award