**Conhecer a Plataforma**

Conheça a plataforma, entenda como a arquitetura funciona, como as chamadas do sistema são tratadas, o gerenciamento de memória, saiba como o tratamento de exceções protege sua aplicação, enfim todo o possível relacionado a arquitetura que está desenvolvendo, sendo uma linguajem compilada ou interpretada não menos importante é o entendimento do funcionamento da arquitetura só assim será possível criar um programa de qualidade, o cuidado em conhecer o em plataforma física ou virtual.

**Estude os recursos da linguagem**

Estude os recursos da linguagem que está utilizando, conheça as bibliotecas padrões oferecidas, os recursos da ferramenta e como você pode tirar proveito de todas as facilidades disponíveis. A produtividade depende do domínio deste conjunto de habilidades, mesmo que você consiga resolver 90% das situações com os comandos comuns da linguajem os outros 10% podem ficar mal resolvidos sem os conhecimentos específicos criados para que a linguagem busque ser o mais completa possível.

**Conheça outras linguagens**

Conheça outras linguagens, não importa o quanto bom você é em uma ou quantas linguagens conhece, sempre terá algo novo que pode lhe surpreender, por isso fique atento e dedique um tempo para buscar novas linguagens, conhecer e discutir sobre elas. Com as novas linguagens vêm novas técnicas criadas por experientes no assunto, conhecendo novas técnicas você aperfeiçoa seus conhecimentos e de alguma forma sempre tira proveito.

**Estude outros códigos**

Estude outros códigos, exemplos disponíveis na internet, distribuições abertas, outros aplicativos de sua empresa, enfim todo tipo de código que tiver acesso, procure entender a arquitetura o porquê foi feito daquela maneira e quais as técnicas utilizadas pelo autor. Se for diretamente relacionado à linguagem que trabalha tente entender o código por inteiro, caso sinta muita dificuldade significa que deve estudar mais.

**Leia e estude diversas literaturas**

Leia diversas literaturas e leia mesmo do começo ao final não pule páginas porque acha o assunto desinteressante ou cansativo, isso fará com que seu aprendizado seja superficial, reduzindo a capacidade de compreender o assunto, a leitura de todo o conteúdo faz com que armazene conhecimento maciço sobre o assunto e irá se lembrar quando surgir a necessidade de utilizar certa técnica ou explicação publicada no livro.

**Escreva definitivamente**

Escreva definitivamente, não pense em escrever só para o teste, então após isso comece a corrigir os erros apresentados, ou até mesmo escrever trechos de código para depois verificar se vai dar certo mas sim, planeje, pense o que vai escrever e escreva para funcionar, isso pode evitar horas de transtorno e correção de código, se adotar esse padrão com o tempo vai alcançar uma nível de erros mínimo reduzindo tempo de desenvolvimento e aumentando a confiabilidade de seu código.

**Faça o tratamento de erros**

Faça o tratamento de erros, teste o retorno de funções utilize tratamento de exceções e busque minimizar a possibilidade de erro passar despercebida, pode ser em uma chamada padrão que dificilmente pode gerar erro, se existe a possibilidade de ocorrer erro, deixe o código prevendo, pequenos tratamentos podem fazer a diferença na qualidade da aplicação.

**Adote um padrão de programação**

Adote um padrão de programação, de identação, de nomenclatura de variáveis, bibliotecas, funções e arquivos. Use um padrão existente ou defina o seu baseado em vários estudados, nem sempre um padrão existente vai atender as necessidades do projeto, porém, é fundamental a adoção de um modelo de programação, faça o projeto inteiro baseado no padrão assim se torna organizado e entendível com o tempo.

**Controle alterações**

Controle alterações e aprenda a utilizar uma ferramenta de versionamento, basicamente elas servem para que você possa fazer regressão a qualquer mudança do código, permitindo comparar o que foi alterado e assim entender o que aconteceu, irá parecer desperdício de tempo no início, mas pode salvar você de horas de retrabalho quando for corrigir algo que deu errado. Diversas ferramentas trabalham independente de sua plataforma de desenvolvimento, você simplesmente aponta a pasta em que está o projeto e manda fazer o backup/commit.

**Levantar requisitos**

Levantar requisitos e documente o que será feito antes de começar a fazer, isto também poderá lhe salvar horas de programação e retrabalho, se o projeto for pequeno escreva em uma lista corrida de tópicos todos os recursos que estarão presentes no sistema e faça o cliente assinar, os recursos mais complexos você pode detalhar para que sejam clarificados, se o projeto for complexo utilize alguma técnica de levantamento de requisitos, mas lembre-se, sair programando sem ter definido o que será feito pode vir a ser muito frustrante e arriscado.