1. Especificação do programa:
   1. Objetivos e requisitos. (Farei)
   2. Especificação, por exemplo use-cases. (Não Farei)

É importante mencionar a aplicação e dizer os modelos que você pretende usar para validar a implementação (isso é use-cases).

1. Projeto do programa:
   1. Arquitetura. (Não Farei)
   2. Critérios de projeto utilizados. (Não Farei)
   3. Diagramas de arquitetura e/ou segmentação do programa, por exemplo UML. (Não Farei)
   4. Organização do programa (componentes, módulos, classes, ...), por exemplo, diagramas de classe UML. (Farei)
   5. Diagramas de organização dos dados, por exemplo diagramas de modelagem de dados, ou entidade relacionamentos. (Não Farei)

O importante é fazer uma projeto da implementação. Voce pode escrever textualmente o projeto e apresentar diagramas de módulos e classes.

1. Código fonte:
   1. Comentário inicial de cada módulo, identificando o autor. (Farei)
   2. Comentários, cabeçalho de módulos, classes e funções. (Farei)
   3. Assertivas para dados e procedimentos. (Farei)
   4. Pseudo instruções. (Farei)
   5. Procure estabelecer ou adotar padrões de programação. (Farei)
2. Roteiro de teste efetuado, composto de:
   1. Critérios de teste utilizados. (Farei)
   2. Descrição dos casos de teste. (Farei)
   3. Na medida do possível procure utilizar testes automatizados. (Farei)
   4. Scripts de teste automatizado. (Farei)
   5. Logs gerados pelo teste automatizado. (Farei)
3. Documentação para o usuário. (Não Farei)

Tem que fazer a documentação para o usuário. Isso é simples, não custa nada.