Requirements

1) Think of a logical data model that includes two entities that have a cardinality relationship of 1 to 1..*. Each one of the entities in your logical model should have at least 2 attributes of varying type.
2) Create a REST based API that implements all the CRUD operations using a resource-based modeling approach for each of the entities. This API should use JSON as its default serialization.
3) Apply a security layer to the REST API of your choosing.
4) Create a simple self-hosted UI that utilizes the full feature set of your API design.
5) The REST API should write its supported entities to a local persistence mechanism of your choosing. Ideally the persistence mechanism should be able to be packaged with the project.
6) Each invocation of your REST API should write a small log message describing a given interaction to the following AMQP message broker. The body of each message published to the message broker should include an "email" attribute with a value set to your email address.

| | |
|---|---|
| IP Address | 34.206.111.76 |
| Port | 5672 |
| Username | Callminer |
| Password | KxYOWbVfog |
| Exchange Name | amq.direct |
| Channel Name | cm.callminer |

7) Use standard UML notation to document your models.
8) Use standard UML notation to create a logical component diagram of your system.
9) Create a solutions design document that includes all the UML artifacts and provides an explanation of each of the major components in your system.

Assumptions

1) The implementation of your system should be written in a recent version of the .NET framework
2) The implementation of your system should adhere to the SOLID design principles
3) The project should have a composition root and use dependency injection to facilitate object creation.
4) The project should utilize layering to appropriately separate the different application concerns.
5) Application code should be DRY

Delivery Instructions

After you have created your solution and its supporting documentation please do the following:

1) Package all system artifacts, including the source code for your solution into a zip file
2) Create an MD5 checksum of the zip file
3) Upload the artifacts to a remote file storage service such as AWS S3, Drop Box etc.
4) Email an externally accessible link to the zip file and corresponding MD5 checksum to tracy.ryan@callminer.com