



Course: Introduction to Software Engineering

Lab Assignment 02

Name: Trương Samuel

StudentID: 20127610

Class: 20CLC11

Lecturers: Nguyễn Thị Minh Tuyên

Hồ Tuấn Thanh

Nguyễn Lê Hoàng Dũng

Contents

I. Introduction..... 3

II. Evidences 3

1. API..... 3

2. Commit..... 5

I. Introduction

Role code: Back-end

Language: Go

Framework: Echo, GORM

[Tutorial link](#)

II. Evidences

1. API

```
// POST API
func CreateStudent(c echo.Context) error {
    db := storage.GetDBInstance()
    temp := &model.Students{}
    db.Last(&temp)
    student := &model.Students{
        Id: temp.Id + 1,
    }

    if err := c.Bind(student); err != nil {
        return err
    }

    db.Create(&student)
    return c.JSON(http.StatusCreated, student)
}
```

Figure 1. Create API

```

// GET API
func GetAllStudents(c echo.Context) error {
    db := storage.GetDBInstance()
    students := []model.Students{}

    if err := db.Find(&students).Error; err != nil {
        return c.JSON(http.StatusNotFound, "Not found student")
    }

    return c.JSON(http.StatusOK, students)
}

func GetStudent(c echo.Context) error {
    db := storage.GetDBInstance()
    student := &model.Students{}

    // Check if exist
    id, _ := strconv.Atoi(c.Param("id"))
    if err := db.Take(&student, id).Error; err != nil {
        return c.JSON(http.StatusNotFound, "Not found student")
    }

    return c.JSON(http.StatusOK, student)
}

```

Figure 2. Get API

```

// PUT API
func UpdateStudent(c echo.Context) error {
    db := storage.GetDBInstance()
    student := &model.Students{}

    // Check if exist
    id, _ := strconv.Atoi(c.Param("id"))
    if err := db.Take(&student, id).Error; err != nil {
        return c.JSON(http.StatusNotFound, "Not found student")
    }

    if err := c.Bind(student); err != nil {
        return err
    }
    student.Id = id

    db.Save(student)
    return c.JSON(http.StatusOK, student)
}

```

Figure 3. Update API

```

// DELETE API
func DeleteStudent(c echo.Context) error {
    db := storage.GetDBInstance()
    student := &model.Students{}

    // Check if exist
    id, _ := strconv.Atoi(c.Param("id"))
    if err := db.Take(&student, id).Error; err != nil {
        return c.JSON(http.StatusNotFound, "Not found student")
    }

    db.Delete(student, id)
    return c.JSON(http.StatusOK, student)
}

```

Figure 4. Delete API

2. Commit

