

ПРИЛОЖЕНИЕ А

Исходный код веб-приложения

```
namespace BLL.Concrete
{
    public class UserSecurityService : EmailService, IUserSecurityService
    {
        public UserSecurityService(IUserRepository userRepository, IDbContextScopeFactory
dbContextScopeFactory) : base(userRepository, dbContextScopeFactory) { }
        public UserSecurityService(IUserRepository userRepository, IDbContextScopeFactory
dbContextScopeFactory, string emailRegularExpression) : base(userRepository, dbContextScopeFactory,
emailRegularExpression) { }

        #region IUserSecurityService

        public bool ValidateUser(string email, string password, IEqualityComparer<string>
passwordComparer)
        {
            UserExceptionsHelper.GetEmailExceptions(email, this.emailValidationRegex);
            UserExceptionsHelper.GetPasswordExceptions(password);
            bool result = false;
            using (var context = dbContextScopeFactory.CreateReadOnly())
            {
                var user = this.repository.GetUserByEmail(email);
                if (user != null && user.IsApproved)
                {
                    result = passwordComparer.Equals(user.Password, password);
                }
            }
            return result;
        }

        public bool ChangePassword(string email, string oldPassword, string newPassword,
IEqualityComparer<string> passwordComparer)
        {
            UserExceptionsHelper.GetEmailExceptions(email, this.emailValidationRegex);
            UserExceptionsHelper.GetPasswordExceptions(oldPassword, "oldPassword");
            UserExceptionsHelper.GetPasswordExceptions(newPassword, "newPassword");
            bool result = false;
            using (var context = dbContextScopeFactory.Create())
            {
                var user = this.repository.GetUserByEmail(email);
                if (user != null)
                {
                    if (passwordComparer.Equals(user.Password, oldPassword))
                    {
                        user.Password = newPassword;
                        this.repository.Update(user);
                        result = true;
                    }
                }
                context.SaveChanges();
            }
            return result;
        }
    }

    #endregion
}
```